

Національний технічний університет України
«Київський політехнічний інститут»

А. П. Мовчан, О. В. Степанець

МЕТОДИ СТАТИЧНОЇ ОПТИМІЗАЦІЇ

Навчальний посібник

2012

Мовчан А.П.

Навчальний посібник: Методи статичної оптимізації. Навч. посіб. / Мовчан А.П., Степанець О.В. — К.: НТУУ «КПІ», 2012. — 138 с.

Викладені методи вирішення задач статичної оптимізації як у загальних випадках, так їх застосування розробці та оптимізації систем автоматичного керування. Висвітлені методи одновимірної оптимізації функцій, включаючи градієнтні та методи нульового порядку, класичні підходи до вирішення задач умовної оптимізації, методи багатовимірної оптимізації та лінійного програмування.

Для студентів вищих навчальних закладів, які навчаються за напрямом «Автоматизація та комп'ютерно-інтегровані технології». Також може бути корисним для інженерно-технічних працівників відповідних спеціальностей.

Зміст

ВСТУП.....	5
1. ПОСТАНОВКА ЗАДАЧ СТАТИЧНОЇ ОПТИМІЗАЦІЇ.	
КЛАСИФІКАЦІЯ МЕТОДІВ ОПТИМІЗАЦІЇ ФУНКЦІЙ.....	6
1.1. Основні поняття та визначення.....	6
1.2. Приклади формулювання задач оптимізації	8
1.3. Геометрична інтерпретація задачі статичної оптимізації.	14
1.4. Класифікація методів розв'язання задач оптимізації функцій.	16
1.5. Основні поняття математичного програмування.	19
1.6. Контрольні завдання	24
2. МЕТОДИ КЛАСИЧНОГО АНАЛІЗУ ДЛЯ РОЗВ'ЯЗКУ ЗАДАЧ	
УМОВНОЇ ОПТИМІЗАЦІЇ.....	26
2.1. Метод виключення	26
2.2. Метод множників Лагранжа.	27
2.3. Умови Куна-Такера	34
2.4. Контрольні завдання	38
3. МЕТОДИ ОДНОМІРНОЇ ОПТИМІЗАЦІЇ	40
3.1. Метод Фібоначчі.	40
3.2. Метод золотого перетину.....	42
3.3. Методи поліноміальної інтерполяції.	44
3.4. Метод послідовного оцінювання із використанням квадратичної	
апроксимації.....	44
3.5. Метод пошуку з використанням кубічної апроксимації.....	47
3.6. Застосування методів одновимірної оптимізації в задачах ідентифікації	
об'єктів управління.....	50
3.7. Застосування методів одновимірної оптимізації в задачах адаптивного	
управління.....	51
3.8. Контрольні завдання	54
4. ГРАДІЄНТНІ МЕТОДИ БАГАТОМІРНОЇ ОПТИМІЗАЦІЇ	56
4.1. Методи першого порядку	56
4.1.1. Градієнтний метод.....	56
4.1.2. Метод найшвидшого спуску.	58
4.1.3. Метод крутого спуску	62
4.1.4. Метод спряжених градієнтів (Флетчера-Рівса)	63
4.2. Методи другого порядку	64
4.2.1. Метод Ньютона	64
4.2.2. Модифікований метод Ньютона	66
4.2.3. Метод Девідона-Флетчера-Пауела (ДФП).	66
4.3. Застосування градієнтних методів в задачах адаптивного управління.	68
4.4. Контрольні завдання	73

5. МЕТОДИ НУЛЬОВОГО ПОРЯДКУ БАГАТОМІРНОЇ ОПТИМІЗАЦІЇ	75
5.1. Метод Гауса-Зейделя	76
5.2. Метод Розенброка	78
5.3. Метод пошуку по симплексу	84
5.4. Метод Хука-Дживса	91
5.5. Методи випадкового пошуку	98
5.5.1. Послідовність випадкових чисел	99
5.5.2. Сліпий пошук	100
5.5.3. Метод випадкових напрямків	101
5.6. Методи оптимізації при наявності обмежень	102
5.6.1. Методи штрафних функцій	102
5.6.2. Метод проекції градієнта	106
5.7. Контрольні завдання	112
6. ЛІНІЙНЕ ПРОГРАМУВАННЯ	115
6.1. Постановка задачі	115
6.2. Нормальна форма задачі ЛП	115
6.3. Властивості лінійних задач оптимізації	117
6.4. Симплексний метод розв'язку задач ЛП	120
6.4.1. Основи симплексного методу	120
6.4.2. Алгоритм симплекса-методу в табличній формі	122
6.5. Розв'язок задач на мінімум	125
6.6. Вибір початкового допустимого базисного розв'язку	125
6.7. Приклади постановок задач ЛП	126
7. РЕАЛІЗАЦІЯ СТАТИЧНОЇ ОПТИМІЗАЦІЇ В СИСТЕМАХ КЕРУВАННЯ	128
7.1. Оптимізація процесу за допомогою математичних моделей	128
7.1.1. Оптимізація на основі методів розпізнавання ситуації	128
7.1.2. Декомпозиційна оптимізація	131
7.2. Контрольні завдання	136
Список використаної та рекомендованої літератури	138

ВСТУП

У практичній діяльності часто зустрічаються задачі оптимізації, у яких змінні стану та керування не залежать від часу. Такі задачі називаються статичними. Розв'язок задачі статичної оптимізації забезпечує знаходження незалежних змінних, оптимальних з погляду якихось прийнятих умов. Прикладами таких задач оптимізації можна назвати:

- задачі вибору сталих режимів технологічних процесів;
- задачі оперативного керування, зокрема оптимального розподілу матеріальних і енергетичних ресурсів, планування ремонтів і перевезень;
- задачі вибору оптимальних параметрів окремих виробів, настройок регуляторів оптимального проектування систем керування.

Математичною основою розв'язання таких задач є методи оптимізації функцій.

Ціль навчального посібника – ознайомлення з методикою формалізації оптимізаційних задач і способами їх розв'язання.

Основна увага приділяється методам і алгоритмам, які використовуються в інженерній практиці при дослідженні, проектуванні, експлуатації, аналізі функціонування об'єктів і систем керування. Розглядаються методи оптимізації, орієнтовані на розв'язок задач з неперервними змінними і обмеженнями та з однією цільовою функцією, тобто математичні методи, часто об'єднані в рамках теорії нелінійного і лінійного програмування. Розглянуто використання методів оптимізації в задачах ідентифікації і оптимізації технологічних процесів, оптимізації настройок систем керування, побудови функціонально адаптивних систем регулювання.

В посібнику наведено приклади і вправи, які рекомендується використовувати при самостійній роботі, а також задачі, які допоможуть викладачу при проведенні семінарських занять і розробці завдань для самостійної роботи студентів.

В основу навчального посібника покладені методичні розробки і матеріали лекційного курсу «Моделювання і оптимізація систем керування», який автор викладає на протязі ряду років на кафедрі автоматизації теплоенергетичних процесів НТУУ «КПІ», а також результати одержані при виконанні науково дослідних робіт.

Автори висловлюють щире подяку аспіранту Бунке О.С. та студентці Грасевич Ю.О. за допомогу при підготовці рукопису.

1. ПОСТАНОВКА ЗАДАЧ СТАТИЧНОЇ ОПТИМІЗАЦІЇ. КЛАСИФІКАЦІЯ МЕТОДІВ ОПТИМІЗАЦІЇ ФУНКЦІЙ

1.1. Основні поняття та визначення

У будь-якій задачі статичної оптимізації доводиться оперувати наступними основними елементами. По-перше, наборами незалежних змінних (керуючих параметрів) $u_i (i=1,2,\dots,m)$, які утворюють m -мірний вектор незалежних змінних \vec{u} . По-друге, наборами залежних змінних (вихідних величин) $y_i (i=1,2,\dots,n)$, які утворюють n -мірний вектор залежних змінних \vec{y} , до того ж вважаючи, що матриця спостерігача C у виразі $\vec{y} = C \cdot \vec{x}$, де \vec{x} – вектор змінних стану, є одиничною, будемо надалі оперувати з n -мірним вектором \vec{x} . По-третє, деяким функціональним виразом I , що включає в себе розглянуті змінні, і повинен бути мінімізований або максимізований. Цей вираз називають *цільовою функцією* (функцією мети), *критерієм оптимальності* або *показником якості* і записують у вигляді:

$$I = f_0(\vec{x}, \vec{u}). \quad (1.1)$$

Цільова функція є скалярною мірою ефективності знайденого розв'язку задачі.

Постановка будь-якої задачі оптимізації включає в себе умови, які характеризують прийнятні значення змінних, і називаються обмеженнями задачі. Обмеження містять у собі рівняння зв'язку між залежними та незалежними змінними у вигляді рівнянь а також функціональні та параметричні обмеження у вигляді нерівностей. Обмеження у вигляді рівнянь записуються в такий спосіб:

$$f_j(\vec{x}, \vec{u}) = 0, \quad j = 1, 2, \dots, m_1. \quad (1.2)$$

Аналогічно, обмеження у вигляді нерівностей мають вигляд:

$$G_j(\vec{x}, \vec{u}) \leq 0, \quad j = m_1 + 1, \dots, m_2. \quad (1.3)$$

Будь-яке обмеження у вигляді нерівності за допомогою введення додаткової змінної z_j можна звести до еквівалентного обмеження типу рівняння: $G_j(\vec{x}, \vec{u}) + z_j^2 = 0$.

З урахуванням вищесказаного, задачу статичної оптимізації можна сформулювати в такий спосіб: мінімізувати (або максимізувати) критерій (1.1)

$$I = f_0(\vec{x}, \vec{u}) \rightarrow \min_{\vec{u}}(\vec{x}, \vec{u}), \quad (1.4)$$

за умови:

$$\begin{aligned} f_j(\vec{x}, \vec{u}) &= 0, j = 1, 2, \dots, m_1, \\ G_j(\vec{x}, \vec{u}) &\leq 0, j = m_1 + 1, \dots, m_2, \end{aligned}$$

скорочено можна записати:

$$I_{\min} = \min_{\vec{u}} \{ f_0(\vec{x}, \vec{u}) \mid f_j(\vec{x}, \vec{u}) = 0, j = 1, \dots, m_1; G_j(\vec{x}, \vec{u}) \leq 0; j = m_1 + 1, \dots, m_2 \}. \quad (1.5)$$

Іноді зустрічається інший вигляд запису постановки задачі:

$$I = f_0(\vec{x}, \vec{u}) \rightarrow \min_{\vec{u}}; \quad \vec{x} \in D_x, \quad \vec{u} \in D_u, \quad (1.6)$$

де D_x і D_u – області допустимих розв'язків для змінних \vec{x} і \vec{u} відповідно, що визначаються умовами (1.2), (1.3), тобто:

$$D_x = \{ \vec{x} : f_j(\vec{x}, \vec{u}) = 0, j = 1, \dots, m_1; G_j(\vec{x}, \vec{u}) \leq 0; j = m_1 + 1, \dots, m_2 \} \quad (1.7)$$

$$D_u = \{ \vec{u} : f_j(\vec{x}, \vec{u}) = 0, j = 1, \dots, m_1; G_j(\vec{x}, \vec{u}) \leq 0; j = m_1 + 1, \dots, m_2 \} \quad (1.8)$$

Необхідно відзначити, що задача мінімізації функції $f_0(\vec{x}, \vec{u})$ еквівалентна задачі максимізації функції $-f_0(\vec{x}, \vec{u})$, а обмеження $G_j(\vec{x}, \vec{u}) \leq 0$ еквівалентні обмеженням $-G_j(\vec{x}, \vec{u}) \geq 0$. Це дозволяє замість задачі мінімізації розв'язувати задачу максимізації та навпаки.

Розв'язок статичної задачі оптимізації дозволяє визначити оптимальну точку $(\vec{x}_{\text{opt}}, \vec{u}_{\text{opt}})$ в області допустимих розв'язків D_x і D_u , у зв'язку з чим задачу статичної оптимізації можна сформулювати трохи інакше: знайти значення незалежних змінних $\vec{u} = (u_1, u_2, \dots, u_m)$ і відповідні їм значення залежних змінних $\vec{x} = (x_1, x_2, \dots, x_m)$, що забезпечують мінімальне (максимальне) значення критерію оптимальності $f_0(\vec{x}, \vec{u})$ при виконанні умов (1.2) і (1.3).

Необхідно підкреслити, що в результаті розв'язання статичної задачі оптимізації визначається оптимальна точка у просторі розв'язків $(\vec{x}_{\text{opt}}, \vec{u}_{\text{opt}})$, у той час як у випадку динамічної задачі оптимізації визначаються функції від часу $\vec{x}_{\text{opt}}(t)$ і $\vec{u}_{\text{opt}}(t)$.

В залежності від числа незалежних змінних і наявності або відсутності обмежень розрізняють наступні задачі оптимізації.

Якщо число незалежних змінних (керуючих параметрів) більше одного ($m \geq 2$), то вона називається *багатопараметричною* (*багатомірною*) задачею оптимізації, а при $m = 1$ ця задача буде називатись *однопараметричною* (*одномірною*) задачею оптимізації.

При відсутності обмежень (1.2) і (1.3) задача називається задачею *безумовної оптимізації*, у протилежному випадку – задачею *умовної оптимізації*. Необхідно відзначити, що задачі однопараметричної оптимізації завжди є задачами умовної оптимізації, тому що пошук

мінімуму цільової функції здійснюється на деякому інтервалі $[u^-, u^+]$. Пошук мінімуму цільової функції на нескінченній числовій осі u позбавлений змісту.

Інша класифікація задач, пов'язана із методами їхнього розв'язання, буде наведена далі.

1.2. Приклади формулювання задач оптимізації

Правильному формулюванню сприяє використання стандартної послідовності основних етапів формалізації. Ця послідовність така:

1. Змістовна (словесна) постановка задачі;
2. Введення позначень для змінних (бажано із розмірностями);
3. Запис у прийнятих позначеннях критерію оптимальності, як функції від усіх або частини шуканих змінних;
4. Виділення множини допустимих значень змінних, яку визначають як обмеження, накладені на кожну із змінних (автономні обмеження), і умови, накладені на сукупність змінних. Наявність останніх змушує враховувати при формулюванні задачі не тільки ті змінні, які входять в критерій оптимальності, але всі так чи інакше з ними пов'язані.

Задача 1. Вибір режиму печі

Змістовна постановка задачі.

У печі (рис. 1.1, а) спалюється паливо. Потрібно вибрати такі значення витрат палива і повітря, для яких температура в печі була б максимальною. При конкретному складі палива і стані печі температура визначається їх співвідношенням.

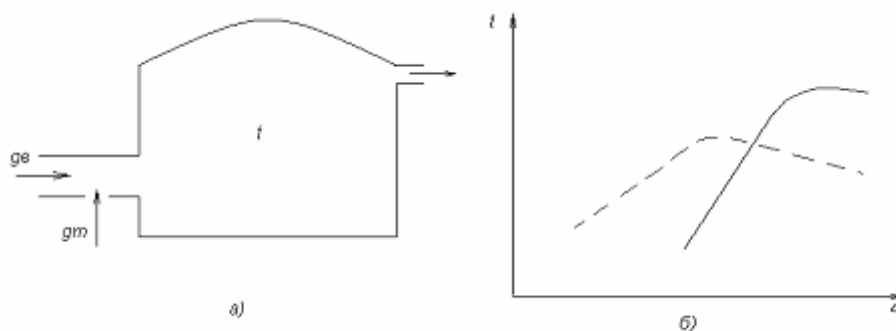


Рис. 1.1. Спалювання палива у печі

Введемо позначення: g_m — витрата палива, g_e — витрата повітря. Температуру в печі, що залежить від g_m, g_e , позначимо $t = f_0(g_e, g_m)$ (рис. 1.1, б).

Критерій оптимізації:

$$f_0(\alpha) \rightarrow \max_{\alpha}, \quad (1.9)$$

Обмеження на змінні величини задачі:

$$g_{\dot{o}. \max} \geq g_{\dot{o}} \geq 0; g_{\dot{a}. \max} \geq g_{\dot{a}} > 0; g_{\dot{a}} / g_{\dot{o}} \geq \alpha, \quad (1.10)$$

де $g_{\dot{o}. \max}$, $g_{\dot{a}. \max}$ – гранично допустимі значення цих змінних, α – відношення витрати повітря до витрати палива.

Відмітимо відразу, що наведена постановка правильна лише у першому наближенні. В дійсності на вигляд функції f_0 впливає цілий ряд факторів не врахованих при постановці задачі. До таких факторів відносяться температура повітря, температура і склад палива, стан пальників. Так що залежність $t = f_0(\alpha)$ може мати різну форму (пунктирна і суцільна лінії на рис. 1.1, б), до того ж деякі з цих факторів практично важко виміряти. Таким чином, мова може йти, наприклад, про максимум математичного очікування f_0 при заданих витратах палива і повітря.

Задача 2. Вибір оптимального складу палива

Змістовна постановка задачі.

Тепловій електричній станції (ТЕС) потрібне вугілля із вмістом сірки, не більшим 0,03% і золи - не більшим 3,25%. Маємо три сорти вугілля:

Таблиця 1

Сорт	Вміст сірки	Зольність	Ціна за 1т, грн.
А	0,06	2,0	30
В	0,04	4,0	30
С	0,02	3,0	45

Знайти, в якому співвідношенні потрібно змішати вугілля А, В і С, щоб задовольнити обмеженням на вміст домішок сірки і попелу, та мінімізувати ціну.

Формулювання критерію оптимізації.

Нехай 1 т суміші містить вугілля типу А, В і С відповідно, u_1 , u_2 і u_3 . Тоді ціна однієї тонни може бути знайдена за допомогою наступного математичного виразу:

$$S = 30 \cdot u_1 + 30 \cdot u_2 + 45 \cdot u_3 \quad (1.11)$$

Критерій оптимальності. Цільова функція в задачі приймає вигляд:

$$S(u_1, u_2, u_3) \rightarrow \min_{u_1, u_2, u_3} \quad (1.12)$$

Область допустимих розв'язків визначається наступними співвідношеннями:

$$u_1 + u_2 + u_3 = 1, \quad (1.13)$$

$$0,06 \cdot u_1 + 0,04 \cdot u_2 + 0,02 \cdot u_3 \leq 0,03 \cdot 10^{-2}, \quad (1.14)$$

$$2 \cdot u_1 + 4 \cdot u_2 + 3 \cdot u_3 \leq 3,25 \cdot 10^{-2}. \quad (1.15)$$

Задача 3. Вибір розміщення центрального вузла при прокладці трас

Змістовна постановка задачі.

Задано розміщення деяких споживачів сировини або енергії. Потрібно так вибрати розміщення центрального вузла проміжного джерела, з якого постачають усіх споживачів, щоб сумарна довжина трубопроводів (рис. 1.2) була мінімальною.

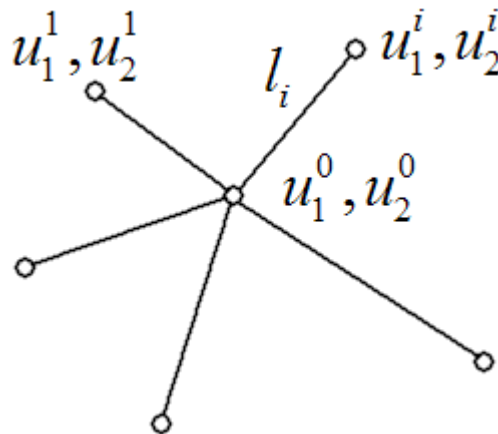


Рис. 1.2. Схема розміщення центрального вузла і споживачів

Введення позначень для змінних.

Позначимо через u_1^0 та u_2^0 координати центрального вузла на площині, а через u_1^i , u_2^i координати i -го споживача. Відстань від центрального вузла до i -го споживача l_i можна знайти за допомогою співвідношення:

$$l_i = \sqrt{(u_1^0 - u_1^i)^2 + (u_2^0 - u_2^i)^2}. \quad (1.16)$$

Тепер можна записати критерій оптимальності:

$$I = \sum_i l_i = \sum_i \sqrt{(u_1^0 - u_1^i)^2 + (u_2^0 - u_2^i)^2} \rightarrow \min_{u_1^0, u_2^0} \quad (1.17)$$

Що стосується множини допустимих рішень, то в даному випадку допустимі усі дійсні значення змінних u_1^0 та u_2^0 , так як обмеження у задачі відсутні.

Задача 4. Задача оптимального проектування

а). *Змістовна постановка.* Вибрати довжину твірної і діаметр днища циліндра таким чином, щоб при заданому об'ємі загальна довжина зварних швів виявилась мінімальною. При цьому потрібно врахувати, що

зварювати доводиться краї зігнутого в циліндр листа по твірній і до отриманої труби приварити днища (рис.1.3).

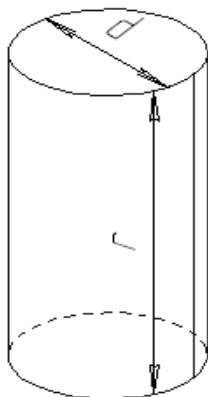


Рис. 1.3. До задачі оптимального проектування

Позначення для змінних: l і d – довжина твірних і діаметр циліндра; V – його об’єм.

Критерій оптимальності – загальна довжина зварених швів. У прийнятих позначеннях він запишеться так:

$$I = 2\pi d + l \rightarrow \min_{l,d}. \quad (1.18)$$

Множина допустимих розв’язків (будемо називати її множиною D) визначається системою обмежень типу нерівностей і рівнянь:

$$d \geq 0; l \geq 0; \quad (1.19)$$

$$\frac{\pi d^2 l}{4} - V = 0 \quad (1.20)$$

При цьому умови (1.19) накладені автономно на кожну із шуканих змінних, а умова (1.20) пов’язує їх одна з одною.

б). Змістовна постановка. Спроекувати ємність, яка має форму циліндра, об’єм якого рівний $20,0 \text{ м}^3$. Визначити розміри ємності, при яких вартість його виготовлення мінімальна.

Математичне формулювання. Для оцінки вартості виготовлення знайдемо повну поверхню циліндру, що визначає кількість матеріалу, яка піде на його виготовлення, а також довжину звареного шва.

Тоді вартість ємності можна оцінити за виразом:

$$D = F \cdot a_F + L \cdot a_L \quad (1.21)$$

де F – площа повної поверхні ємності; L – загальна довжина зварних швів; a_F, a_L – вартість одиниці площі матеріалу і вартість виготовлення одиниці зварного шва.

Позначимо висоту циліндру через H , радіус днища – R , які не повинні виходити за деякі рамки:

$$R_{\min} \leq R \leq R_{\max} \quad (1.22)$$

$$H_{\min} \leq H \leq H_{\max} \quad (1.23)$$

У математичному вигляді сформульовану задачу можна записати таким чином:

$$I = F \cdot a_F + L \cdot a_L \rightarrow \min_{R, H}, \quad (1.24)$$

$$V = \pi R^2 H = 20, \quad (1.25)$$

де V – об'єм ємності.

$$F = 2\pi R H + 2\pi R^2 \quad (1.26)$$

$$L = 2 \cdot 2\pi R + 2H \quad (1.27)$$

$$R_{\min} \leq R \leq R_{\max} \quad (1.28)$$

$$H_{\min} \leq H \leq H_{\max} \quad (1.29)$$

Ми отримали математичну модель для знаходження оптимального розв'язку. Вона складається із критерію оптимальності, обмежень типу рівнянь і обмежень типу нерівностей.

Задача 5. Задача оптимального параметричного синтезу регулятора

Змістовна постановка.

Необхідно знайти оптимальні параметри настройок пропорційно-інтегрального регулятора (ПІ-регулятора) для об'єкту першого порядку із запізненням, які б забезпечили мінімальне значення квадратичного інтегрального критерію якості при ступінчатому збуренні по завданню.

Аналіз задачі і введення позначень.

Структурна схема системи автоматичного регулювання зображена на рис. 1.4:

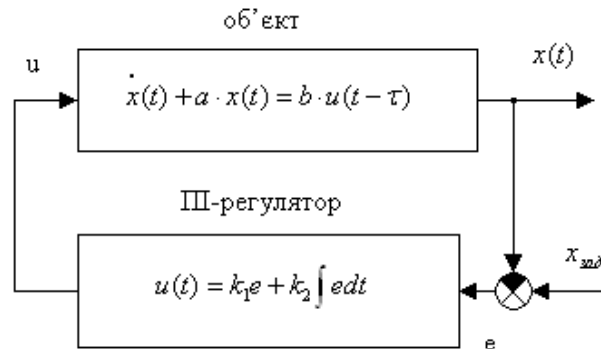


Рис. 1.4. Структурна схема системи автоматичного регулювання

Прийняті позначення:

$x(t)$ – вихідна змінна об'єкту регулювання;

$u(t)$ – регулююча змінна (керування);

τ – час запізнення;

a, b – параметри об'єкту;

$e = x_{зад} - x(t)$ – помилка регулювання;

k_1, k_2 – коефіцієнти ПІ-регулятора, які повинні бути знайдені із умов досягнення мінімуму інтегрального критерію оптимізації.

Математична постановка задачі.

Критерій оптимізації має наступний вигляд:

$$I = f_0(x, u) = \int_{t_n}^{t_k} e^2 dt \rightarrow \min_{k_1, k_2}. \quad (1.30)$$

Множина допустимих розв'язків повинна задовольняти наступним співвідношенням:

- обмеженню типу рівняння – математичний опис об'єкту регулювання

$$f_1(x, u) = x(t) + a \cdot x(t) - b \cdot u(t - \tau) = 0 \quad (1.31)$$

із граничними умовами:

$$x(t_n) = x_n, \quad x(t_k) = x_k, \quad (1.32)$$

де $x(t_n)$, $x(t_k)$ – значення вихідної величини в початковий t_n і кінцевий t_k моменти часу.

- Обмеженню типу рівняння – математичний опис ПІ-регулятора

$$f_2(x, u) = u - k_1 \cdot e + k_2 \int e dt. \quad (1.33)$$

- Обмеженням типу нерівність

$$G_1(u) = u_{\min} - u \leq 0; \quad G_2(u) = u_{\max} - u \geq 0 \quad (1.34)$$

або $u_{\min} \leq u \leq u_{\max}$, які обумовлені обмеженою пропускнуою здатністю регулюючого органу.

Оптимальне проектування систем

Під задачею проектування розуміється створення нового об'єкта чи системи, що матимуть задані властивості чи характеристики. Об'єктами проектування можуть бути технічні системи, в тому числі системи автоматичного управління, економічні системи тощо. Основні проблеми, що виникають при рішенні задачі проектування, пов'язані з завданням структури проектного об'єкта (структурний синтез), а також із вибором параметрів в рамках уже відомої структури (параметричний синтез). Основні вимоги до створюваного об'єкта задаються вектором вихідних величин \vec{y} .

Задача проектування об'єкта в ряді випадків може бути представлена як задача рішення системи нерівностей, що називаються специфікаціями. Ця система має такий вигляд:

$$y_i(\vec{x}) \leq t_i, \quad i = \overline{1, m}, \quad (1.35)$$

де \vec{x} – невідомий вектор параметрів проектування, що підлягає вибору.

Під задачею оптимального проектування розуміють задачу отримання вектора \bar{x} з умови

$$\bar{x} = \arg \min_x I(x), \quad (1.36)$$

де $I(x)$ – деякий критерій оптимальності функціонування системи.

Таким чином, задачу оптимального проектування можна представити як задачу умовної оптимізації (1.36) з обмеженнями типу нерівності (1.35).

1.3. Геометрична інтерпретація задачі статичної оптимізації.

Для наочного зображення задачі і роботи алгоритмів оптимізації використовують геометричну інтерпретацію в площині незалежних змінних.

Нехай задача оптимізації функції $f_0(\bar{u})$, яка містить обмеження типу рівнянь і нерівностей має вигляд:

$$I = f_0(\bar{u}) = (u_1 - 2)^2 + (u_2 - 2)^2 \rightarrow \min_{\bar{u}}; \quad (1.37)$$

$$f_1(\bar{u}) = u_1 + u_2 = 1; \quad (1.38)$$

$$G_1(\bar{u}) = u_1 \leq 2; \quad (1.39)$$

$$G_2(\bar{u}) = u_2 \leq 2; \quad (1.40)$$

$$G_3(\bar{u}) = u_1 \geq 0; \quad (1.41)$$

$$G_4(\bar{u}) = u_2 \geq 0. \quad (1.42)$$

Цільова функція зображується за допомогою ліній постійного рівня, для побудови яких, наприклад, у площині двох змінних необхідно прирівнювати цільову функцію до одного із обраних постійних значень c_1, c_2, c_3 і т.д. і, задаючи значення однієї незалежної змінної (u_1), визначати значення іншої (u_2).

Графічно це зображується лінією, в окремому випадку замкнутої, у площині незалежних змінних u_1 і u_2 (рис. 1.5).

Подібним же чином графічно інтерпретуються й обмеження. Обмеження-рівняння типу (1.2) зображуються лініями, обмеження-нерівності (1.3) позначають області припустимих розв'язків.

На рис. 1.5 області за заштрихованими сторонами ліній $G_j(u_1, u_2)$ відповідають областям недопустимих розв'язків відповідно до умов (1.3). Очевидно, якщо розв'язується задача безумовної оптимізації $\min f_0(\bar{u})$ або задача умовної оптимізації

$$\min \{ f_0(\bar{u}) \mid G_j(\bar{u}) \leq 0, j = \overline{1,4} \}, \quad (1.43)$$

то розв'язок задачі буде відповідати точці безумовного оптимуму $\vec{u}_{ir\delta}^{\delta}$, і при цьому виконуються співвідношення: $\min f_0(\vec{u}) < c_3 < c_2 < c_1$. Якщо ж розв'язується задача умовної оптимізації

$$\min \left\{ f_0(\vec{u}) \mid G_j(\vec{u}) \leq 0, j = \overline{1,4}; f_j(\vec{u}) = 0, j = 1 \right\}, \quad (1.44)$$

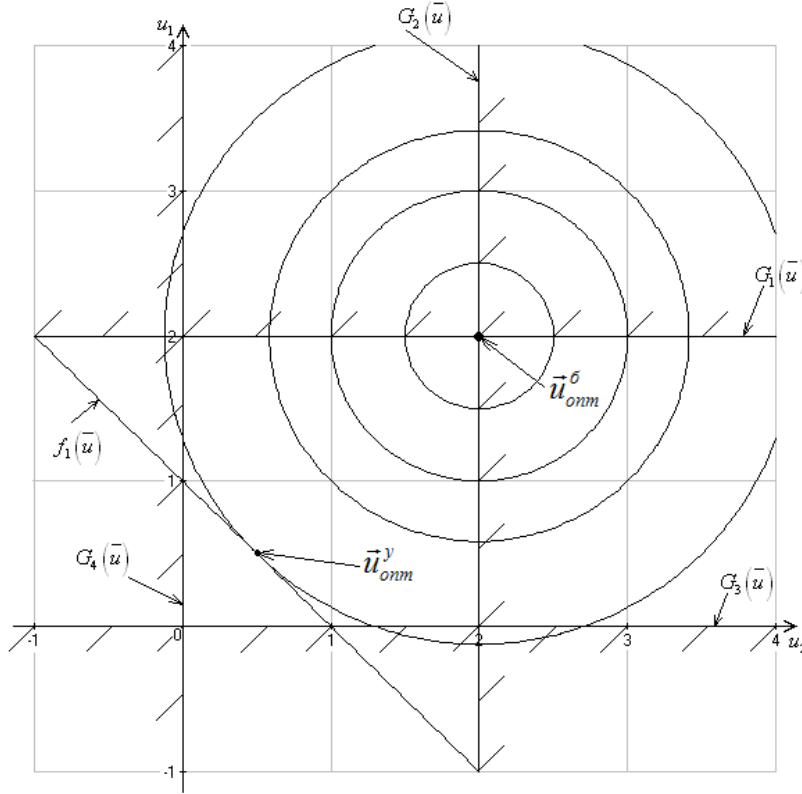


Рис. 1.5. Графічна інтерпретація задачі (1.37)

тобто з обмеженням типу рівності, то розв'язок задачі буде знаходитися в точці умовного оптимуму $\vec{u}_{ir\delta}^{\delta}$.

Розглянемо інший приклад. Нехай задача оптимізації функції $f_o(\vec{u})$, яка містить обмеження типу рівнянь і нерівностей має вигляд:

$$I = f_o(\vec{u}) = (u_1 - 2)^2 + (u_2 - 2)^2 \rightarrow \min_{u_1, u_2} \quad (1.45)$$

$$f_1(\vec{u}) = u_1 - 2 \cdot u_2 = 0, \quad (1.46)$$

$$G_1(\vec{u}) = u_1 + u_2 - 2 > 0, \quad (1.47)$$

$$G_2(\vec{u}) = u_2 - 1 > 0, \quad (1.48)$$

$$G_3(\vec{u}) = u_2 - 4 < 0, \quad (1.49)$$

$$G_4(\vec{u}) = u_1 - 5 < 0, \quad (1.50)$$

$$G_5(\vec{u}) = u_1 > 0. \quad (1.51)$$

Графічну інтерпретацію цієї задачі зображено на рис. 1.6.

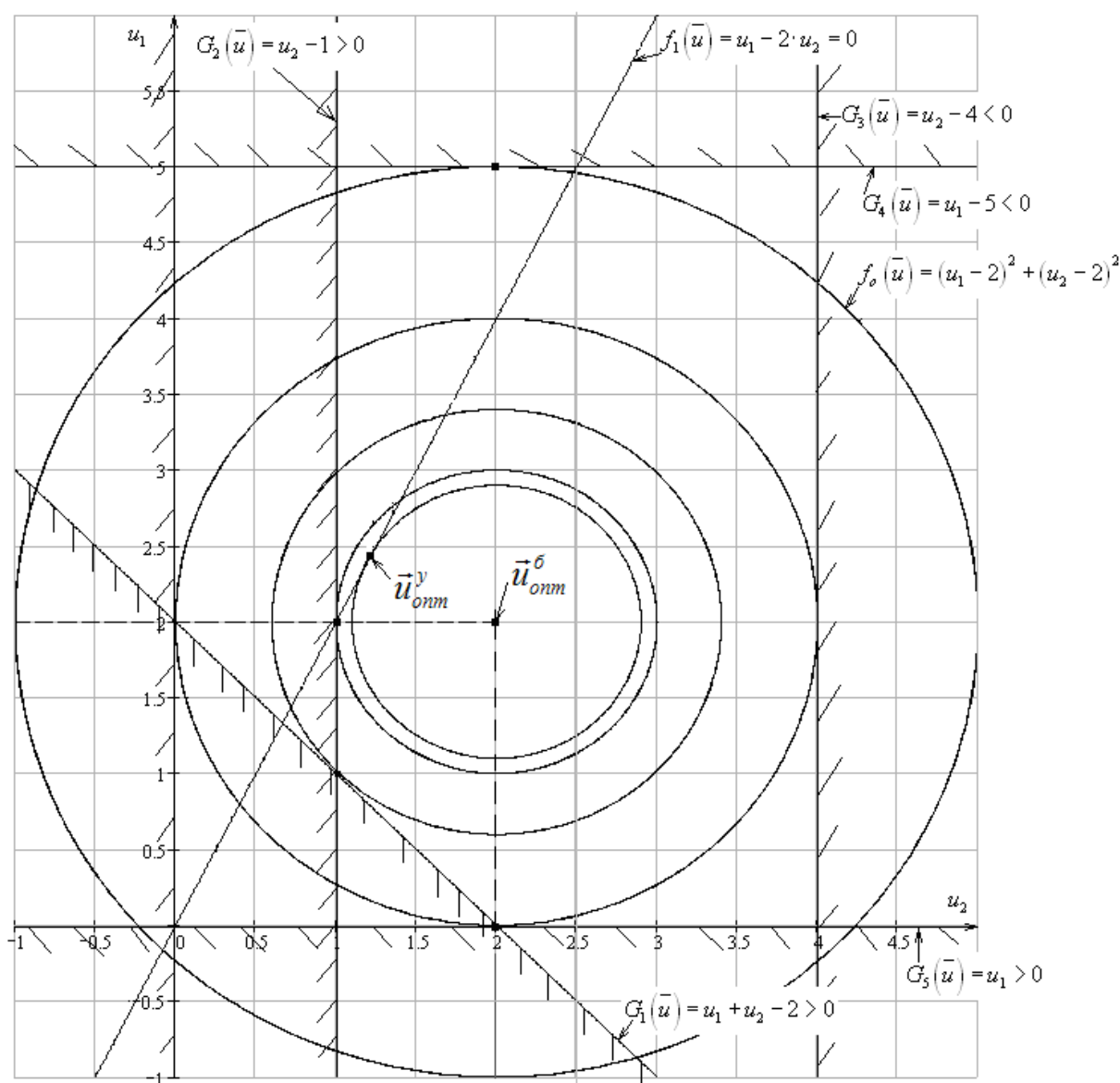


Рис. 1.6. Графічна інтерпретація задачі (1.45)

Геометрична інтерпретація задачі доцільна тільки у просторі двох незалежних змінних, тобто при $m = 2$.

1.4. Класифікація методів розв'язання задач оптимізації функцій.

Задачі оптимізації відрізняються великою різноманітністю та складністю, у зв'язку з чим відсутній універсальний метод їхнього рішення.

Усі методи розв'язання оптимізаційних задач можна розбити на дві великі групи: методи класичного аналізу та методи математичного програмування.

Методи, розроблені в класичній математиці, засновані на розв'язанні систем алгебраїчних рівнянь, отриманих при прирівнюванні нулеві частинних похідних досліджуваної функції по оптимізуючих параметрах. Ці методи дозволяють розв'язувати задачі пошуку оптимуму тільки при відсутності обмежень на оптимізовані параметри або при обмеженнях у вигляді рівностей (метод множників Лагранжа).

Методи класичного аналізу застосовують у тому випадку, коли аналітичний вигляд критерію оптимізації й обмежень відомий, і рівняння можна виразити в явній формі.

Область математики, що досліджує питання теорії і методи розв'язання неklasичних задач оптимізації, одержала назву *математичного програмування (МП)*.

Ця назва пов'язана із тим, що розв'язання неklasичних оптимізаційних задач здійснюється за допомогою визначеної програми дій, тобто алгоритму, що вказує послідовність операцій, у зв'язку з чим ці методи іноді називають також *алгоритмічними*.

Методи МП класифікують відповідно до типів неklasичних оптимізаційних задач (задач МП). Виділяють наступні класи методів і задачі МП:

1. *Нелінійне програмування (НП)*. Задача НП повинна мати хоча б одне нелінійне обмеження або нелінійну цільову функцію відносно незалежних змінних \vec{u} .

У класі задач нелінійного програмування виділяють задачі квадратичного програмування.

Для розв'язку загальної задачі нелінійного програмування розроблена велика кількість алгоритмів, ефективність роботи яких залежить від природи цільової функції $f_0(\vec{u})$. Всі ці алгоритми можна певним чином класифікувати.

Залежно від постановки задачі оптимізації розрізняють:

1. *Методи безумовної оптимізації.*
2. *Методи умовної оптимізації.*

Залежно від здатності алгоритму шукати локальний або глобальний екстремум розрізняють:

1. *Методи локальної оптимізації.*
2. *Методи глобальної оптимізації.*

Залежно від числа незалежних змінних задачі оптимізації розрізняють:

1. *Методи одомірної оптимізації.*

2. *Методи багатомірної оптимізації.*

Залежно від характерних рис функціонування алгоритму розрізняють:

1. *Методи нульового порядку (методи, які не використовують похідні цільової функції).*

2. *Методи першого порядку (методи, що використовують перші похідні цільової функції).*

3. *Методи другого порядку (методи, що використовують другі похідні цільової функції).*

Необхідно відзначити, що багато алгоритмів нульового і першого порядку при розв'язанні задач багатомірної оптимізації використовують методи одномірної оптимізації, які й будуть розглянуті в першу чергу.

Квадратичне програмування. У задачах квадратичного програмування всі обмеження лінійні, а цільова функція квадратична, тобто має вигляд:

$$f_0(u_1, u_2, \dots, u_n) = \sum_{i=1}^m c_i u_i + \sum_{\substack{i,j=1, \\ i \leq j}} u_i d_{i,j} u_j \quad (1.52)$$

або у векторній формі:

$$f_0(\vec{u}) = \vec{c} \vec{u} + \vec{u}^T D \vec{u} \quad (1.53)$$

де d_{ij} - елементи симетричної матриці D.

2. Лінійне програмування (ЛП). У задачах ЛП всі обмеження і цільова функція є лінійними.

3. Цілочисельне програмування (ЦП). У задачах ЦП незалежні змінні приймають тільки цілі значення.

4. Стохастичне програмування (СП). У задачах СП вихідна інформація містить елементи невизначеності, або деякі параметри задачі носять випадковий характер.

5. Геометричне програмування. У задачах геометричного програмування цільова функція й обмеження є поліноміальними функціями

$$f(\vec{u}) = \sum_{i=1}^n \alpha_i u_i^{a_{i1}} u_2^{a_{i2}} \dots u_m^{a_{im}} \quad (1.54)$$

Відзначимо, що задачі геометричного програмування є окремим класом задач НП і узагальненням окремого випадку задач НП - задач квадратичного програмування.

Нижче будуть розглянуті методи, що найчастіше застосовуються в інженерній практиці, а саме: нелінійне і лінійне програмування, методи класичного аналізу.

1.5. Основні поняття математичного програмування.

Опуклість є одним з найбільш важливих математичних понять, яке використовується у математичному програмуванні, у зв'язку з тим, що для задач МП особливий інтерес представляють опуклі множини.

Множина точок, що утворюють область D , називається *опуклою*, якщо будь-які дві точки, що належать цій множині, можуть бути з'єднані відрізком прямої лінії, який також цілком належить цій множині.

У математичній формі визначення опуклої множини дається в такий спосіб: точкова множина (множина точок) D_u називається *опуклою*, якщо для будь-яких двох точок $\vec{u}_1, \vec{u}_2 \in D_u$ будь-яка лінійна комбінація \vec{u}_1 і \vec{u}_2 також відноситься до D_u , тобто

$$\vec{u}^* = \alpha \vec{u}_1 + (1 - \alpha) \vec{u}_2 \in D_u, \quad 0 \leq \alpha \leq 1 \quad (1.55)$$

Останній вираз є формою опису всіх можливих відрізків прямої, що з'єднують \vec{u}_1 і \vec{u}_2 . Прикладами опуклих множин є: відрізок, пряма, гіперплощина, куля. На рисунку 1.7 зображені приклади опуклих і не опуклих множин.

Аналогічно визначається поняття опуклої функції. Функція $f(\vec{u})$ є опуклою на опуклій множині D_u , якщо для будь-яких двох точок $\vec{u}_1, \vec{u}_2 \in D_u$, і для усіх $\alpha, 0 \leq \alpha \leq 1$ справедлива формула:

$$f[\alpha \vec{u}_1 + (1 - \alpha) \vec{u}_2] \leq \alpha f(\vec{u}_1) + (1 - \alpha) f(\vec{u}_2) \quad (1.56)$$

Якщо в (1.56) буде строга нерівність (тобто знак «<») при $0 \leq \alpha \leq 1$, то функція $f(\vec{u})$ буде називатися *строго опуклою*.

Якщо в (1.56) знак нерівності змінити на протилежний, то $f(\vec{u})$ буде називатися *увігнутою* (*строго увігнута* при строгій нерівності). Звідси випливає, що якщо $f(\vec{u})$ опукла (строго опукла), то $-f(\vec{u})$ увігнута (строго увігнута).

Для опуклих множин характерно те, що перетин будь-якого числа опуклих множин є опуклою множиною.

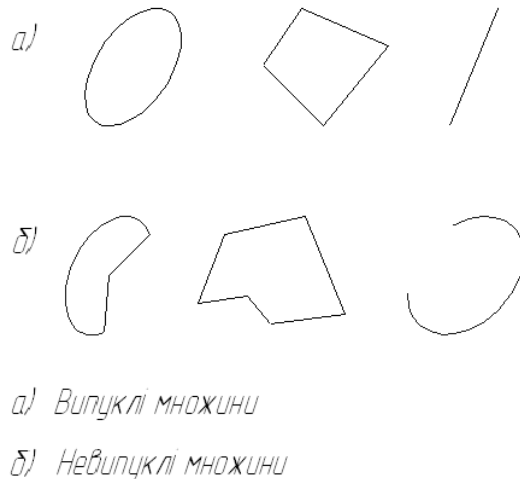


Рис. 1.7. Випуклі та невикуклі множини

У зв'язку із введенням поняття опуклості можна доповнити класифікацію методів і задач МП, опуклим програмуванням (ОП). Задача мінімізації опуклої функції $f(\vec{u})$, заданої на опуклій множині D_u , називається *задачею опуклого програмування*. До задач опуклого програмування можуть відноситися задачі як нелінійного, так і лінійного програмування.

Важливою властивістю задач ОП є те, що оптимальний розв'язок задачі досягається у локальному мінімумі, який є, в той же час, і глобальним.

Вектор \vec{u}^{opt} називається *точкою локального (відносного) мінімуму* функції $f(\vec{u})$ на множині D_u , якщо при деякому достатньо малому $\varepsilon > 0$ для всіх $\vec{u} \neq \vec{u}^{opt}$ і $\vec{u} \in D_u$, що задовольняють умові $|\vec{u} - \vec{u}^{opt}| \leq \varepsilon$, виконано нерівність $f(\vec{u}^{opt}) \leq f(\vec{u})$, тобто, якщо в ε – околиці точки \vec{u}^{opt} функція $f(\vec{u})$ не приймає значення меншого ніж $f(\vec{u}^{opt})$.

Вектор \vec{u}^{opt} називається *точкою глобального (абсолютного) мінімуму* функції $f(\vec{u})$ на множині D_u , якщо $\vec{u}^{opt} \in D_u$ і $f(\vec{u}^{opt}) < f(\vec{u})$ для всіх $\vec{u} \in D_u$, $\vec{u} \neq \vec{u}^{opt}$, тобто якщо в жодній іншій точці області допустимих рішень (допустимої області) D_u функція $f(\vec{u})$ не приймає значення меншого ніж $f(\vec{u}^{opt})$.

Очевидно, що глобальний мінімум – це найменший з усіх локальних мінімумів. Задача оптимізації, у якій цільова функція має в допустимій області один локальний мінімум, називається *одноекстремальною (унімодальною)* задачею оптимізації. Якщо локальних мінімумів декілька, то задача називається *багате́кстремальною (мультимодальною)*.

Багатоекстремальні задачі називаються *задачами неопуклого програмування*.

Допустимий розв'язок.

Будь-який розв'язок, тобто будь-який вектор \vec{u} , що належить області допустимих розв'язків D_u , яка обумовлена обмеженнями у вигляді рівнянь і нерівностей (1.19) і (1.20), називається *допустимим розв'язком* задачі оптимізації. Будь-який розв'язок поза D_u називається *недопустимим розв'язком*.

Оптимальний розв'язок (оптимум) є найкращим (мінімальний або максимальний, тобто екстремальний) із усіх допустимих у змісті прийнятої цільової функції.

Якщо локальний оптимум знаходиться на границі допустимої області, то він називається *умовним оптимумом*.

Необхідні і достатні умови оптимальності.

Розглянемо їх для багатомірних задач безумовної оптимізації.

Необхідна умова оптимальності (екстремуму цільової функції): для того, щоб у точці \vec{u}^{onm} функція $f(\vec{u})$ мала безумовний локальний екстремум (тобто приймала найменше або найбільше значення), необхідно щоб усі її частинні похідні в точці \vec{u}^{onm} дорівнювали нулю:

$$\left. \frac{\partial f(\vec{u})}{\partial u_i} \right|_{\vec{u}=\vec{u}^{onm}} = 0, i = 1, 2, \dots, m \quad (1.57)$$

Ця умова називається також *умовою стаціонарності*, тому що в результаті розв'язання системи (1.57) визначаються всі стаціонарні (у яких можливе існування екстремуму або перегину) точки.

Умова стаціонарності може бути записана також наступним чином:

$$\text{grad } f(\vec{u}^{onm}) = \nabla f(\vec{u}^{onm}) = \begin{pmatrix} \frac{\partial f(\vec{u}^{onm})}{\partial u_1} \\ \vdots \\ \frac{\partial f(\vec{u}^{onm})}{\partial u_m} \end{pmatrix} = 0 \quad (1.58)$$

Останній вектор називається *градієнтом* функції $f(\vec{u})$ в точці \vec{u}^{onm} . Необхідно відзначити, що градієнт існує для безперервних і диференційованих функцій. Доведено, що градієнт скалярної функції спрямований у бік найшвидшого збільшення функції (найшвидшого підйому) і ортогональний дотичній до лінії рівня $f(\vec{u}_1)$, що проходить

через дану точку \vec{u}_1 . Вектор, протилежний цьому градієнтові (антиградієнт), спрямований у бік найшвидшого спуску.

Умова (1.57) і еквівалентна йому (1.58) не є достатньою, тому що не визначає характер стаціонарної точки, тобто не відповідає на питання, чи знаходиться в ній максимум або мінімум.

Достатня умова оптимальності: для того щоб двічі безперервно диференційована функція $f(\vec{u})$ мала в стаціонарній точці \vec{u}^{onm} безумовний локальний мінімум (максимум), необхідно, щоб матриця її других похідних була ненегативно (непозитивно) визначеною, і досить, щоб вона була позитивно (негативно) визначеною.

Якщо матриця других похідних функції є невизначеною в стаціонарній точці, то ця точка є *сідловою* точкою цієї функції.

Матриця других похідних називається також *матрицею Гессе* (*гессіаном*) і записується як:

$$H = \nabla^2 f(\vec{u}^{onm}) = \begin{vmatrix} \frac{\partial^2 f(\vec{u}^{onm})}{\partial u_1^2} & \dots & \frac{\partial^2 f(\vec{u}^{onm})}{\partial u_1 \cdot \partial u_m} \\ \vdots & & \\ \frac{\partial^2 f(\vec{u}^{onm})}{\partial u_n \cdot \partial u_1} & \dots & \frac{\partial^2 f(\vec{u}^{onm})}{\partial u_m^2} \end{vmatrix} \quad (1.59)$$

Матриця H є позитивно визначеною тоді і тільки тоді, коли квадратична форма $x^T H x > 0$ для всіх $x \neq 0$, де $x^T = [x_1, x_2, \dots, x_n]$.

Матриця H є ненегативно визначеною тоді і тільки тоді, коли квадратична форма $x^T A x \geq 0$ для всіх x та існує вектор $x \neq 0$, такий, що $x^T A x = 0$.

Матриця H є негативно визначеною тоді і тільки тоді, коли матриця $-H$ є позитивно визначеною, тобто коли $x^T H x < 0$ для всіх $x \neq 0$.

Матриця H є непозитивно визначеною тоді і тільки тоді, коли матриця $-H$ є ненегативно визначеною, тобто коли $x^T H x \leq 0$ для всіх $x \neq 0$.

Матриця H є невизначеною, якщо $x^T H x$ може приймати як додатні, так і від'ємні значення.

Перевірку знаковизначеності матриць можна здійснити, наприклад, за допомогою *критерію Сильвестра*. Якщо a_{ij} є елементами симетричної матриці розмірності $m \times m$ (1.59), то для позитивної визначеності необхідно і достатньо виконання наступних нерівностей:

$$a_{11} > 0, \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} > 0, \dots, \begin{vmatrix} a_{11} & \dots & a_{1m} \\ \vdots & & \vdots \\ a_{m1} & \dots & a_{mm} \end{vmatrix} > 0 \quad (1.60)$$

Для ненегативної визначеності необхідно і достатньо:

$$a_{11} \geq 0, \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} \geq 0, \dots, \begin{vmatrix} a_{11} \dots a_{1m} \\ \vdots \\ a_{m1} \dots a_{mm} \end{vmatrix} \geq 0 \quad (1.61)$$

Для негативної визначеності необхідно і достатньо:

$$(-1)^1 \cdot a_{11} > 0, (-1)^2 \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} > 0, \dots, (-1)^m \cdot \begin{vmatrix} a_{11} \dots a_{1m} \\ \vdots \\ a_{m1} \dots a_{mm} \end{vmatrix} > 0 \quad (1.62)$$

Для неопозитивної визначеності необхідно і достатньо:

$$(-1)^1 \cdot a_{11} \geq 0, (-1)^2 \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} \geq 0, \dots, (-1)^m \cdot \begin{vmatrix} a_{11} \dots a_{1m} \\ \vdots \\ a_{m1} \dots a_{mm} \end{vmatrix} \geq 0 \quad (1.63)$$

1.6. Контрольні завдання

1. Дайте визначення опуклої та увігнутої функції.

2. Нехай дана точка опуклої області задовольняє достатнім умовам існування локального мінімуму опуклої функції, визначеної в цій області. Чи є цей мінімум глобальним?

3. Формалізувати приведені нижче змістовні постановки задачі оптимізації:

а) Визначити оптимальне відношення діаметра і висоти циліндричної ємності, що має при заданій поверхні максимальний об'єм;

б) Визначити оптимальне відношення діаметра і висоти циліндричної ємності без верхньої кришки, що має при заданій поверхні максимальний об'єм;

в) Визначити оптимальні діаметри верхньої та нижньої основ, а також висоту конічної ємності без верхньої кришки, що має при заданій площі поверхні максимальний об'єм;

4. Задано наступні функції однієї змінної:

а) $f(x) = x^5 + x^4 - \left(\frac{x^3}{3}\right) + 2$,

б) $f(x) = (2x+1)^2(x-4)$.

Для кожної з даних функцій знайдіть:

- 1) інтервали зростання, спадання;
- 2) точки перегину (якщо такі існують);
- 3) інтервали, в яких функція увігнута, опукла;
- 4) локальні і глобальні максимуми (якщо такі існують);
- 5) локальні і глобальні мінімуми (якщо такі існують).

5. Знайти точки стаціонарності функції та класифікувати їх:

$$f(u_1, u_2) = u_1^2 + u_1 u_2 + u_2^3 + u_1.$$

6. Перевірити, чи має функція

$$f(x, y) = xy + 2x^2 - 4y^2$$

сідлову точку на площині x, y .

7. Обчислити значення цільової функції в точці умовного екстремуму:

$$I(u) = (u - 1)^2 \rightarrow \min_u$$

$$u - 2 \geq 0.$$

8. Обчислити значення цільової функції в точці умовного екстремуму:

$$I(u) = (u - 2)^2 \rightarrow \min_u$$

$$u - 1 \leq 0.$$

9. Обчислити значення цільової функції в точці безумовного екстремуму:

$$I(\vec{u}) = (u_1 - 1)^2 + 2(u_2 - 2)^2 + (u_3 + 1)^2 \rightarrow \min_{\vec{u}}$$

10. В задачі 3а виразити висоту через діаметр і поверхню ємності, звівши тим самим задачу до максимізації функції однієї змінної. Записати необхідні умови її оптимальності і знайти рішення.

11. Зобразити графічно задачу пошуку умовного екстремуму і знайти рішення:

а)

$$I(u_1, u_2) = (u_1 - 2)^2 + (u_2 - 2)^2 \rightarrow \min_{u_1, u_2}$$

$$u_1 - u_2 = 0;$$

$$u_1 \geq 3; u_2 \geq 3.$$

б)

$$I(u_1, u_2) = (u_1 - 2)^2 + 2(u_2 - 1)^2 \rightarrow \min_{u_1, u_2}$$

$$u_1 - 2u_2 = 0;$$

$$u_1 \geq 2; u_2 \geq 1.$$

в)

$$I(u_1, u_2) = (u_1 - 3)^2 + 2u_2^2 \rightarrow \min_{u_1, u_2}$$

$$u_1 - 2u_2 = 0;$$

$$u_1 \geq 4; u_2 \geq 1.$$

2. МЕТОДИ КЛАСИЧНОГО АНАЛІЗУ ДЛЯ РОЗВ'ЯЗКУ ЗАДАЧ УМОВНОЇ ОПТИМІЗАЦІЇ

Як уже говорилося вище, розв'язок задачі оптимізації полягає у знаходженні екстремуму цільової функції в області допустимих рішень.

Існують методи класичного аналізу для пошуку екстремуму як у задачах безумовної, так і умовної оптимізації.

Для розв'язку задачі безумовної оптимізації необхідно розв'язати систему рівнянь (1.58) і в кожній зі знайдених точок перевірити виконання умови негативної або позитивної визначеності матриці Гессе (1.59) та визначити цільову функцію.

Для розв'язку задач умовної оптимізації при наявності обмежень типу рівностей застосовуються методи, засновані на зведенні задач умовної оптимізації до задач безумовної оптимізації.

2.1. Метод виключення

Припустимо, що функції, які визначають обмеження типу рівнянь $f_j(\vec{u})=0$, $j=1,\dots,m_1$ мають в околиці розглянутої точки неперервні частинні похідні до другого порядку по всіх аргументах $j=1,\dots,m_1$, де $m_1 < n$. Тоді, якщо складений із частинних похідних по перших m_1 аргументах, визначник (якобіан) відмінний від нуля

$$\begin{vmatrix} \frac{\partial f_1(\vec{u})}{\partial u_1} & \dots & \frac{\partial f_1(\vec{u})}{\partial u_{m_1}} \\ \vdots & & \vdots \\ \frac{\partial f_{m_1}(\vec{u})}{\partial u_1} & \dots & \frac{\partial f_{m_1}(\vec{u})}{\partial u_{m_1}} \end{vmatrix} \neq 0, \quad (2.1)$$

то система рівнянь $f_j(\vec{u})=0$ має розв'язки відносно u_1,\dots,u_{m_1} і може бути сформульована наступним чином:

$$u_j = g_j(u_{m_1+1}, \dots, u_m), \quad j=1, 2, \dots, m_1 \quad (2.2)$$

Підставляючи останні вирази у вираз для цільової функції, отримаємо задачу пошуку безумовного екстремуму функції $m-m_1$ змінних.

Провести виключення частини компонент вектора \vec{u} буває важко, тому може бути використаний інший метод.

2.2. Метод множників Лагранжа.

Метод множників Лагранжа містить наступні етапи:

1) складається функція $m+m_1$ змінних, яка називається *функцією Лагранжа*:

$$L(\vec{u}, \vec{\lambda}) = f_0(\vec{u}) + \sum_{j=1}^{m_1} \lambda_j f_j(\vec{u}), \quad (2.3)$$

де числа λ_j називаються *множниками Лагранжа*.

2) обчислюються і прирівнюються до нуля її частинні похідні по u_i й λ_j :

$$\begin{cases} \frac{\partial L(\vec{u}, \vec{\lambda})}{\partial u_i} = \frac{\partial f_0(\vec{u})}{\partial u_i} + \sum_{j=1}^{m_1} \lambda_j \cdot \frac{\partial f_j(\vec{u})}{\partial u_i} = 0, \quad i = 1, \dots, m \\ \frac{\partial L(\vec{u}, \vec{\lambda})}{\partial \lambda_j} = f_j(\vec{u}) = 0, \quad j = 1, \dots, m_1 \end{cases} \quad (2.4)$$

3) вирішується система (2.4) $m+m_1$ рівнянь щодо $m+m_1$ невідомих $u_1, u_2, \dots, u_m, \lambda_1, \lambda_2, \dots, \lambda_{m_1}$.

Система рівнянь (2.4) є необхідними умовами екстремуму в задачі умовної оптимізації з обмеженнями типу рівнянь. Як і для задач безумовної оптимізації, необхідні умови не визначають характер стаціонарної точки.

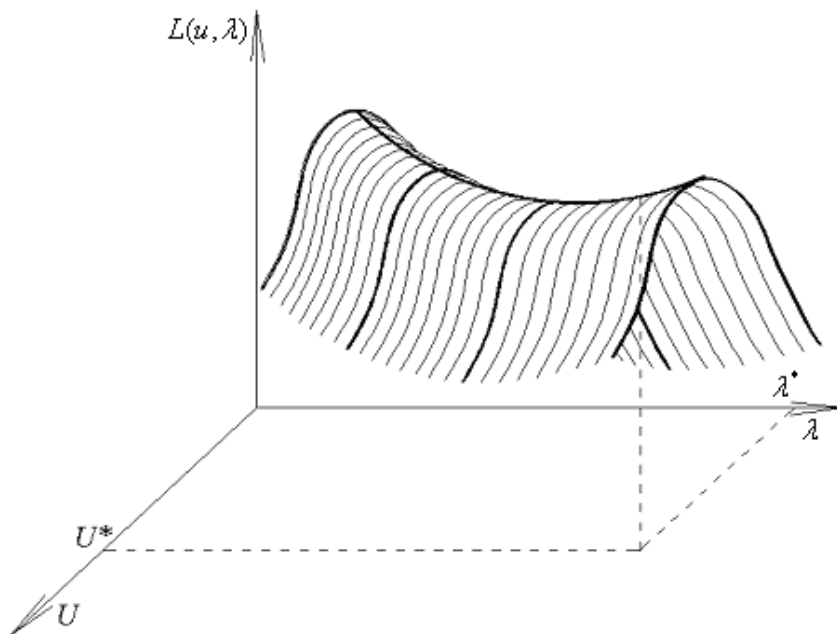


Рис. 2.1. Функція Лагранжа

Можна показати, що екстремум задачі умовної оптимізації з обмеженнями типу рівнянь, досягається при \bar{u} і $\bar{\lambda}$, що відповідають сідловій точці функції Лагранжа (рис. 2.1). Це дозволяє знайти розв'язок задачі методами математичного програмування в тому випадку, коли аналітично розв'язати систему (2.4) важко.

Приклад 2.1. Оптимальне проектування.

Необхідно спроектувати на бетонному фундаменті циліндричний покритий зверху склад, у якому при заданому об'ємі V зовнішня поверхня F повинна бути мінімальною. Діаметр складу позначимо буквою d , а висоту - буквою h .

Математично задача формулюється в такий спосіб:

$$F(d, h) = \frac{\pi d^2}{4} + \pi dh \rightarrow \min, \quad (2.5)$$

при обмеженні:

$$\frac{\pi d^2}{4} h = V \quad (2.6)$$

Функція Лагранжа буде сформульована в наступному вигляді:

$$L(d, h, \lambda) = \frac{\pi d^2}{4} + \pi dh + \lambda(V - \frac{\pi d^2}{4} h) \quad (2.7)$$

З необхідних умов оптимальності:

$$\frac{\partial L}{\partial d} = \frac{\pi d}{2} + \pi h - \lambda \frac{\pi d}{2} = 0, \quad (2.8)$$

$$\frac{\partial L}{\partial h} = \pi d - \lambda \frac{\pi d^2}{4} = 0, \quad (2.9)$$

$$\frac{\partial L}{\partial \lambda} = V - \frac{\pi d^2}{4} h = 0, \quad (2.10)$$

отримують розв'язок:

$$d^* = 2 \cdot \sqrt[3]{\frac{V}{\pi}}, h^* = \sqrt[3]{\frac{V}{\pi}}, \lambda^* = \frac{4}{d^*} \quad (2.11)$$

Мінімальна поверхня становить:

$$F_{\min} = 3 \cdot \sqrt[3]{\pi V^2} \quad (2.12)$$

Приклад 2.2. Оптимальний розподіл ресурсів.

Проблема розподілу ресурсів між декількома паралельно працюючими технологічними лініями має велике практичне значення.

Розглянемо типову постановку задачі.

Дано технологічну систему, що складається з паралельних процесів m , у яких сировина S переробляється в кінцевий продукт P . (рис. 2.2) Залежність виробництва кількості продукту P_i при обробці сировини в

кількості S_i для кожного процесу позначається за допомогою індивідуальної характеристики $P_i(S_i)$. Обмежимо загальну кількість сировини S , що обробляється, заданим значенням S_V .

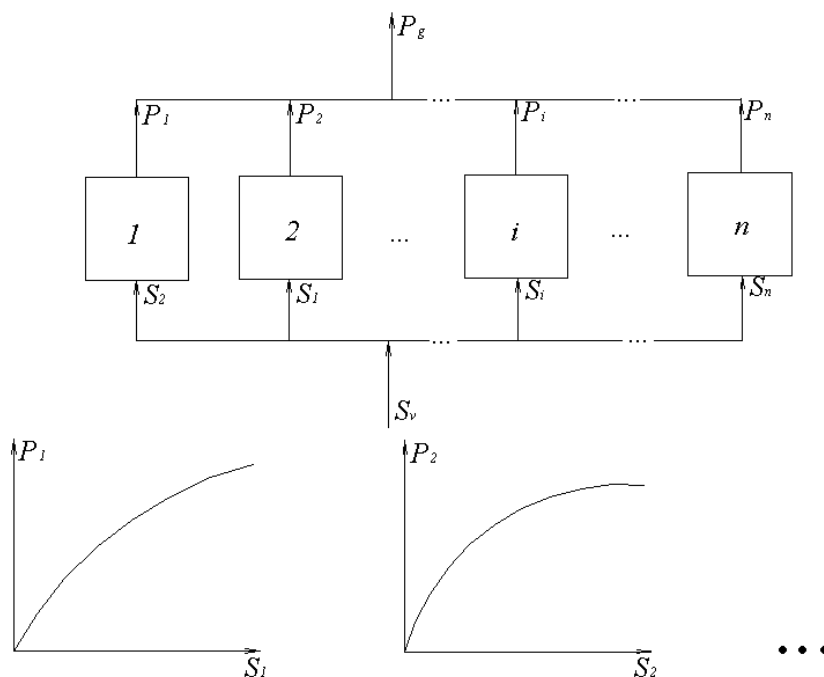


Рис. 2.2. До задачі оптимального розподілу ресурсів

Необхідно знайти таке керування системою (тобто такий розподіл обмежених ресурсів S на всі етапи виробничого процесу), при якому загальне виробництво P було б максимальним:

$$I = \sum_{i=1}^m P_i(S_i) \rightarrow \max \quad (2.13)$$

Очевидно, що:

$$\sum_{i=1}^m S_i = S_V, S_i \geq 0, i = 1, 2, \dots, m \quad (2.14)$$

Складемо функцію Лагранжа:

$$L(S_i, \lambda) = \sum_{i=1}^n P_i(S_i) + \lambda \cdot \left[\sum_{i=1}^m S_i - S_V \right] \quad (2.15)$$

Необхідна умова оптимальності має вигляд:

$$\begin{cases} \frac{\partial L(S_i, \lambda)}{\partial S_i} = \frac{\partial P_i(S_i)}{\partial S_i} + \lambda = 0, i = 1, 2, \dots, m \\ \frac{\partial L(S_i, \lambda)}{\partial \lambda} = \sum_{i=1}^m S_i - S_V = 0 \end{cases} \quad (2.16)$$

Звідси можна знайти невідомі витрати сировини S_i і множник Лагранжа λ .

Необхідно відзначити, що для всіх паралельних процесів необхідне виконання умови:

$$\frac{\partial P_1(S_1)}{\partial S_1} = \frac{\partial P_2(S_2)}{\partial S_2} = \dots = \frac{\partial P_m(S_m)}{\partial S_m}, \quad (2.17)$$

що впливає із системи рівнянь (2.15).

Ця умова виражає загальну закономірність задач розподілу ресурсів. З неї витікає, що малі зміни витрат при переробці сировини на всіх етапах виробничого процесу ведуть до однакових змін продуктивності.

Приклад 2.3. Задача оптимального розподілу навантаження для системи очистки стічних вод

Змістовна постановка.

Дана система будівель для біологічної очистки стічних вод. У її склад входять n відстійників для видалення компонентів забруднень з питомою вагою більшою, ніж у воді. Тяжкі частинки осідають на дно і за допомогою насосів відкачуються. Очищена рідина подається у спільний колектор і відправляється або на зберігання, або на злив. Технологічна схема має вигляд, показаний на рис. 2.3.

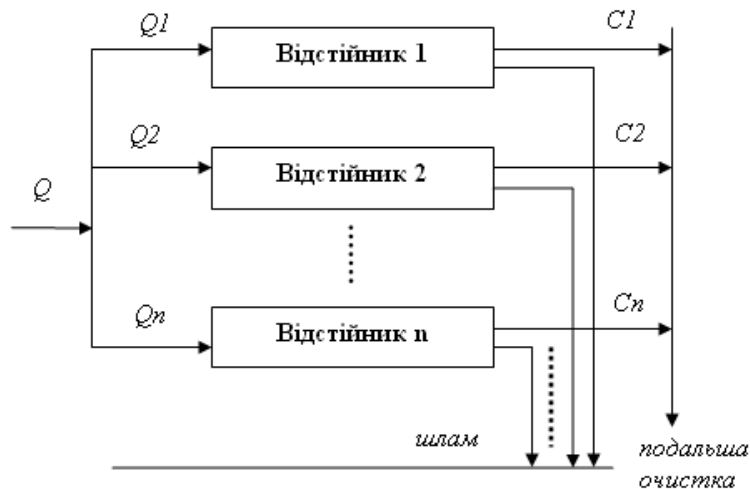


Рис. 2.3. Система очистки стічних вод

Математична постановка задачі.

Задача розподілу навантаження у загальному вигляді може бути записана у наступній формі:

$$I = \sum_{i=1}^n y_i \rightarrow \min_{u_i} \quad (2.18)$$

$$y_i = f_i(u_i, \bar{A}_i) \quad (2.19)$$

$$u_i^{\min} \leq u_i \leq u_i^{\max} \quad (2.20)$$

де u_i – керування, y_i – вихід об'єкту, \bar{A}_i – вектор коефіцієнтів рівняння об'єкту.

Нехай на виході маємо концентрації c_i завислих часток, тоді сумарна кількість забруднень рівна $c_i \cdot Q_i$. Критерій оптимальності можна сформулювати в наступному вигляді:

$$I = \sum_{i=1}^n c_i \cdot Q_i \rightarrow \min_{Q_i} \quad (2.21)$$

Загальне навантаження задане

$$Q_{\text{зад}} = \sum_{i=1}^n Q_i \quad (*)$$

Кожний відстійник може отримувати навантаження в наступних межах:

$$Q_i^{\min} \leq Q_i \leq Q_i^{\max}$$

Математична модель об'єкту керування матиме вигляд: $c_i = \alpha_i Q_i$, де α_i – коефіцієнт, що характеризує роботу відстійника.

Перетворення задачі.

Підставимо c_i з рівняння об'єкту в критерій оптимізації:

$$I = \sum_{i=1}^n \alpha_i \cdot Q_i^2 \rightarrow \min_{Q_i}$$

Позначимо: $x_i = \frac{Q_i}{Q_{\text{зад}}}$

Тоді обмеження типу рівняння (*) матиме вигляд:

$$\sum_{i=1}^n x_i = 1, \quad 0 \leq x_i \leq 1.$$

Розв'язок задачі оптимального розподілу навантаження

При відсутності обмежень типу нерівності задача може бути розв'язана методом множників Лагранжа. Але наявність обмежень типу нерівності потребує введення алгоритму для їх урахування.

Алгоритм рішення задачі може мати наступний вигляд [1]. На першому етапі розв'язку задачі обмеження типу нерівностей не враховуються. Тоді дана задача може бути розв'язана шляхом застосування методу множників Лагранжа (або методом штрафних функцій, але з урахуванням лише обмеження типу рівності).

Коли ця підзадача вирішена, перевіряють виконання обмежень. Якщо вони не порушені, то розв'язок знайдено. В протилежному випадку:

1. Підрахуємо сумарне перевантаження: $\Delta_{\text{перев}} = \sum_j (u_j^{\text{онм}} - u_j^{\text{макс}})$, де $j=1,2,3,\dots$ – перенавантажені агрегати.

2. Визначимо сумарне недовантаження: $\Delta_{недов} = \sum_j (u_j^{\min} - u_j^{opt})$, де $j=1,2,3...$ – недовантажені агрегати.

3. Аналіз ситуації.

- Якщо перенавантаження рівне недовантаженню, то всім недовантаженим агрегатам призначаємо мінімальне навантаження, усім перенавантаженим – максимальне.
- Якщо $\Delta_{перев} < \Delta_{недов}$, то оптимальні навантаження недовантажених агрегатів треба прийняти мінімально допустимими, а решту навантаження розподілити між іншими агрегатами.
- Якщо $\Delta_{перев} > \Delta_{недов}$, то оптимальні навантаження перенавантажених агрегатів треба прийняти максимально можливими, а решту навантаження розподілити між іншими агрегатами.

4. В першому випадку задача вважається розв'язаною, в другому і третьому – здійснюється перехід до кроку 1 з меншим сумарним навантаженням і меншим числом агрегатів.

Приклад 2.4. Оптимальний розподіл навантаження

Маємо три агрегати, залежності продуктивності яких від навантажень описуються функціями

$$\begin{aligned} y_1 &= f_1(u_1) = \sqrt{u_1}; \\ y_2 &= f_2(u_2) = 2\sqrt{u_2}; \\ y_3 &= f_3(u_3) = 3\sqrt{u_3}. \end{aligned} \quad (2.22)$$

Задача розподілу навантаження між цими агрегатами має вигляд:

$$I = \sum_{i=1}^n y_i \rightarrow \max_{u_i}, \quad (2.23)$$

при цьому обмеження на значення вхідних величин наступні:

$$1 \leq u_1 \leq 3; \quad (2.24)$$

$$1 \leq u_2 \leq 3,5; \quad (2.25)$$

$$3 \leq u_3 \leq 5; \quad (2.26)$$

$$\sum_{i=1}^3 u_i = 10. \quad (2.27)$$

Скористаємось алгоритмом призначення граничних навантажень, описаним вище.

Спочатку розв'яжемо задачу оптимізації (2.23) з урахуванням лише обмеження (2.27) за допомогою методу множників Лагранжа. Функція Лагранжа має наступний вигляд:

$$L(\vec{u}, \lambda) = \sqrt{u_1} + 2\sqrt{u_2} + 3\sqrt{u_3} + \lambda(u_1 + u_2 + u_3 - 10).$$

Складаємо систему рівнянь

$$\begin{cases} \frac{1}{2\sqrt{u_1}} + \lambda = 0; \\ \frac{1}{\sqrt{u_2}} + \lambda = 0; \\ \frac{3}{2\sqrt{u_3}} + \lambda = 0; \\ u_1 + u_2 + u_3 = 10. \end{cases} \quad (2.28)$$

Маємо наступні співвідношення

$$u_1 = \frac{1}{4\lambda^2}; \quad u_2 = \frac{1}{\lambda^2}; \quad u_3 = \frac{9}{4\lambda^2}, \quad (2.29)$$

підставивши які у останнє рівняння системи (2.28) отримуємо:

$$\frac{1}{4\lambda^2} + \frac{1}{\lambda^2} + \frac{9}{4\lambda^2} = 10;$$

або

$$\lambda^2 = \frac{14}{40}. \quad (2.30)$$

Підставимо (2.30) у (2.29) і отримаємо рішення задачі (2.23) з обмеженням (2.27):

$$u_1 = \frac{10}{14}; \quad u_2 = \frac{40}{14}; \quad u_3 = \frac{90}{14}; \quad (2.31)$$

Отримане рішення не задовольняє умовам (2.24) і (2.26). Обчислимо сумарне перевантаження і недовантаження

$$\Delta_{перев} = \frac{90}{14} - 5 = \frac{20}{14}; \quad \Delta_{недов} = 1 - \frac{10}{14} = \frac{4}{14}.$$

Оскільки $\Delta_{перев} > \Delta_{недов}$, призначаємо максимальне навантаження перевантаженому агрегату, а решту розподіляємо між першим і другим агрегатами:

$$u_3 = 5$$

$$\sum_{i=1}^3 u_i - u_3 = 10 - u_3 = 10 - 5 = 5$$

Тепер розв'язуємо задачу розподілу навантаження між двома агрегатами. Система рівнянь для методу множників Лагранжа матиме вигляд:

$$\begin{cases} \frac{1}{2\sqrt{u_1}} + \lambda = 0; \\ \frac{1}{\sqrt{u_2}} + \lambda = 0; \\ u_1 + u_2 = 5. \end{cases} \quad (2.32)$$

Розв'язуємо систему (2.32) аналогічно до системи (2.28) і отримуємо наступне рішення:

$$u_1 = 1; \quad u_2 = 4. \quad (2.33)$$

Отримане значення u_2 не задовольняє умові (2.25), тобто маємо перевантаження

$$\Delta_{перев} = 4 - 3,5 = 0,5; \quad \Delta_{недов} = 0.$$

Залишкове навантаження призначається єдиному агрегату, що залишається, тобто першому:

$$u_1 = 5 - 3,5 = 1,5,$$

при цьому обмеження (2.24) не порушується. Таким чином маємо остаточне рішення

$$u_1 = 1,5; \quad u_2 = 3,5; \quad u_3 = 5. \quad (2.34)$$

2.3. Умови Куна-Такера

Задачу умовної оптимізації при наявності обмежень типу рівнянь і нерівностей також можна звести до задачі безумовної оптимізації і застосувати для її розв'язку методи класичного аналізу.

Припустимо, що задача оптимізації сформульована в наступний спосіб. Мінімізувати $f_0(\vec{u})$, при:

$$\begin{aligned} f_j(\vec{u}) &= 0, \quad j = 1, 2, \dots, m_1 \\ G_j(\vec{u}) &\geq 0, \quad j = 1, 2, \dots, m_2 \end{aligned} \quad (2.35)$$

Припустимо, що цільова функція $f_0(\vec{u})$ і допустима область незалежних змінних D_u опуклі. Функції $f_0(\vec{u})$, $f_j(\vec{u})$, $G_j(\vec{u})$ диференційовані на множині $D_u = E_m$, де E_m - евклідів простір.

У цьому випадку умови оптимальності задачі оптимізації (2.35) визначаються наступною теоремою.

Теорема. Для того щоб точка $\vec{u}^{opt} \in D_u$ була оптимальним розв'язком задачі опуклого програмування (2.35) необхідно і достатньо мати в наявності такі вектори $\vec{\lambda}$ і $\vec{\mu}$ функції Лагранжа:

$$L(\vec{u}, \vec{\lambda}, \vec{\mu}) = f_0(\vec{u}) - \sum_{j=1}^{m_1} \lambda_j f_j(\vec{u}) - \sum_{j=1}^{m_2} \mu_j G_j(\vec{u}) \quad (2.36)$$

які задовольняють наступним умовам:

$$\frac{\partial L}{\partial u_i} = 0, i = 1, 2, \dots, m \quad (2.37)$$

$$\frac{\partial L}{\partial \lambda_j} = 0, j = 1, 2, \dots, m_1; \quad (2.38)$$

$$\frac{\partial L}{\partial \mu_j} \leq 0, j = 1, 2, \dots, m_2; \quad (2.39)$$

$$\frac{\partial L}{\partial \mu_j} \cdot \mu_j = 0; \quad (2.40)$$

$$\mu_j \geq 0. \quad (2.41)$$

Розв'язок задач оптимізації із застосуванням умов в аналітичному вигляді можливий лише для досить простих функцій, тому найчастіше доводиться звертатися до чисельних методів розв'язку.

Приклад 2.5. Умови Куна-Такера

Необхідно вирішити наступну задачу оптимізації:

$$f_0(\vec{u}) = (2 - u_1)^2 + (u_2 - 3)^2 \rightarrow \min_{\vec{u}}, \quad (2.42)$$

при обмеженнях:

$$\begin{cases} 3u_1 + 2u_2 - 7 \leq 0; \\ -u_1 + u_2 - 1 \leq 0. \end{cases} \quad (2.43)$$

Функція Лагранжа матиме вигляд:

$$L = (2 - u_1)^2 + (u_2 - 3)^2 - \mu_1(3u_1 + 2u_2 - 7) - \mu_2(-u_1 + u_2 - 1), \quad (2.44)$$

Як умови оптимальності, відповідно до теореми, отримують:

$$\frac{\partial L}{\partial u_1} = -2(2 - u_1) + 3\mu_1 - \mu_2 = 0; \quad (2.45)$$

$$\frac{\partial L}{\partial u_2} = 2(u_2 - 3) + 2\mu_1 + \mu_2 = 0; \quad (2.46)$$

$$\frac{\partial L}{\partial \mu_1} = 3u_1 + 2u_2 - 7 \leq 0; \quad (2.47)$$

$$\frac{\partial L}{\partial \mu_2} = -u_1 + u_2 - 1 \leq 0; \quad (2.48)$$

$$\frac{\partial L}{\partial \mu_1} \mu_1 = \mu_1(3u_1 + 2u_2 - 7) = 0; \quad (2.49)$$

$$\frac{\partial L}{\partial \mu_2} \mu_2 = \mu_2(-u_1 + u_2 - 1) = 0. \quad (2.50)$$

Ця система з урахуванням умов $\mu_1, \mu_2 \leq 0$, вирішується важко. Якщо оптимальний розв'язок знаходиться на краю допустимої області (що в загальному випадку невідомий), можна застосувати наступний евристичний метод:

1) Припустимо, що:

$$\mu_1, \mu_2 > 0. \quad (2.51)$$

2) Відповідно до цього з останніх умов виходить:

$$\begin{cases} 3u_1 + 2u_2 - 7 = 0; \\ -u_1 + u_2 - 1 = 0. \end{cases} \quad (2.52)$$

Звідси:

$$u_1 = 1; u_2 = 2. \quad (2.53)$$

3) Отриманий розв'язок вводимо в рівняння (2.45) і (2.46):

$$-2(2-1) + 3\mu_1 - \mu_2 = 0; \quad (2.54)$$

$$2(2-3) + 2\mu_1 + \mu_2 = 0, \quad (2.55)$$

і визначаємо множники Лагранжа:

$$\mu_1 = \frac{4}{5}, \mu_2 = \frac{2}{5}. \quad (2.56)$$

4) При цьому $u_1 = 1, u_2 = 2$ є оптимальним розв'язком.

Якщо за допомогою описаного алгоритму неможливо знайти розв'язок, то нерівності розв'язуються чисельними методами.

Розглянемо основну задачу опуклого програмування, коли допустима множина D_u має вигляд:

$$D_u = \{u : \vec{G}(\vec{u}) \geq 0, \vec{u} \geq 0\}. \quad (2.57)$$

Тоді умови можна сформулювати за допомогою наступної теореми.

Теорема. Якщо функції $f_0(\vec{u})$ й $\vec{G}(\vec{u})$ основної задачі опуклого програмування

$$f_0(\vec{u}) \rightarrow \min, \quad \vec{u} \in D_u \quad (2.58)$$

безперервно диференційовані на множині $\Gamma_u = \{\vec{u} : \vec{u} \geq 0\}$, то для того щоб пара $\vec{u}^*, \vec{\mu}^*$ була сідловою точкою функції Лагранжа $L(\vec{u}, \vec{\mu}) = f_0(\vec{u}) + \langle \vec{\mu}, \vec{G}(\vec{u}) \rangle$ в області $\vec{u} \geq 0, \vec{\mu} \geq 0$, необхідно і достатньо виконання умов

$$\frac{\partial L^*}{\partial \vec{u}} \geq 0; \quad (2.59)$$

$$\left\langle \vec{u}^*, \frac{\partial L^*}{\partial \vec{u}} \right\rangle = 0; \quad (2.60)$$

$$\vec{u}^* \geq 0; \quad (2.61)$$

$$\frac{\partial L^*}{\partial \vec{\mu}} \leq 0; \quad (2.62)$$

$$\left\langle \vec{\mu}^*, \frac{\partial L^*}{\partial \vec{\mu}^*} \right\rangle = 0 \quad (2.63)$$

$$\vec{\mu}^* \geq 0, \quad (2.64)$$

де:

$$\frac{\partial L^*}{\partial \vec{u}} = \frac{\partial L(\vec{u}, \vec{\mu})}{\partial \vec{u}} \Big|_{\substack{\vec{u}=\vec{u}^* \\ \vec{\mu}=\vec{\mu}^*}}; \quad \frac{\partial L^*}{\partial \vec{\mu}} = \frac{\partial L(\vec{u}, \vec{\mu})}{\partial \vec{\mu}} \Big|_{\substack{\vec{u}=\vec{u}^* \\ \vec{\mu}=\vec{\mu}^*}}. \quad (2.65)$$

Якщо в розглянутій задачі $\Gamma_u = E_m$, то замість перших трьох співвідношень вводиться умова $\frac{\partial L^*}{\partial \vec{u}} = 0$ і ми приходимо до умов аналогічних тим, які записані для задачі (2.35).

2.4. Контрольні завдання

1. Знайдіть найкоротшу відстань від точки $(1, 0)$ до параболи $y^2 = 4x$:

а) шляхом вилучення змінної y ;

б) за допомогою методу множників Лагранжа.

Поясніть, чому процедура (б), на відміну від процедури (а) приводить до отримання розв'язку задачі.

2. Дана наступна задача. Мінімізувати

$$f(\vec{x}) = (x_1 - 3)^2 + (x_2 + 4)^2 + e^{5x_3}$$

при обмеженнях

$$x_1 + x_2 + x_3 \leq 1,$$

$$x_1, x_2, x_3 \geq 0.$$

а) запишіть умови Куна-Такера для цієї задачі.

б) покажіть, що $\vec{x} = (1, 0, 0)$ є точкою оптимуму.

3. Дана наступна задача. Мінімізувати

$$f(\vec{x}) = x_1^2 + x_2^2$$

при обмеженнях

$$(x_1 - 1)^3 - x_2^2 = 0.$$

а) покажіть графічно, що оптимальне рішення має місце при $x_1 = 1, x_2 = 0$.

б) застосуйте метод множників Лагранжа для рішення даної задачі. Поясніть, чому цей метод не приводить до успіху.

4. Покажіть, що функція $f(x, y) = x^2 + y^2$ при обмеженнях $x - y = 5$ досягає мінімуму при $x = 2,5, y = -2,5$.

5. Відкрита коробка, що виготовляється з тонкого листа заліза, має висоту z і прямокутну основу з розмірами x і y . Основи і сторони довжиною x мають товщину d (мала величина), а сторони довжиною y мають товщину $2d$. Якщо кількість матеріалу фіксована, покажіть, що об'єм коробки максимальний при $x=2y=4z$.

6. Запишіть умови Куна-Такера і таким чином розв'яжіть задачу: Мінімізувати функцію

$$f(x, y) = x^2 + y^2$$

при обмеженнях

$$x \geq 0, y \geq 0, x + y \geq 5.$$

7. Оптимізувати розподілення навантаження між трьома енергоблоками; сумарне навантаження $N_{\Sigma} = 850$ МВт, а залежності витрати палива від навантаження кожного окремого енергоблоку визначаються формулами:

$$B_1(N_1) = \alpha N_1^2; \quad B_2(N_2) = 1,1\alpha N_2^2 \quad B_3(N_3) = 1,2\alpha N_3^2,$$

де α – сталий коефіцієнт.

8. Задача умовної оптимізації має вигляд:

$$I(u_1, u_2) = u_1^2 + u_2^2 \rightarrow \min_{u_1, u_2}$$

$$u_1 + u_2 = 4.$$

Записати необхідні умови оптимальності з використанням методу множників Лагранжа і обчислити оптимальне значення множника Лагранжа λ_{opt} .

9. Задача умовної оптимізації має вигляд:

$$I(u_1, u_2) = (u_1 - 2)^2 + u_2^2 \rightarrow \min_{u_1, u_2}$$

$$u_1 - u_2 = 2.$$

Записати необхідні умови оптимальності з використанням методу множників Лагранжа і обчислити оптимальне значення множника Лагранжа λ_{opt} .

3. МЕТОДИ ОДНОМІРНОЇ ОПТИМІЗАЦІЇ

Екстремум одновимірної задачі оптимізації може бути знайдений за допомогою методів класичного аналізу.

У цьому розділі будуть розглянуті деякі чисельні методи одновимірної оптимізації.

3.1. Метод Фібоначчі.

Стратегія пошуку, що лежить в основі методу Фібоначчі ґрунтується на числах Фібоначчі. За їх допомогою визначається ефективний послідовний порядок i -х точок відрізка u , в яких потрібно обчислити значення цільової функції. Числа Фібоначчі F_i утворюються відповідно до простої рекурентної формули.

$$\begin{aligned} F_0 &= F_1 = 1, \\ F_i &= F_{i-1} + F_{i-2}, i = 2, 3, 4, \dots \end{aligned} \quad (3.1)$$

При застосуванні формули (3.1) числа Фібоначчі легко генеруються програмно-технічним шляхом. Доведено, що для визначення екстремального значення функції $f_0(u)$ на відрізку $[a, b]$ з максимальною похибкою $\varepsilon = \frac{b-a}{F_k}$ необхідно визначити принаймні k значень цільової функції $f_0(u)$.

Алгоритм методу Фібоначчі складається з наступних етапів (пошук максимуму):

1. Завдання інтервалу $[a, b]$ й похибки розв'язку ε . Мінімальне число відрізків N знаходиться із виразу:

$$N = \frac{b-a}{\varepsilon} \quad (3.2)$$

2. Визначення числа Фібоначчі F_k , що задовольняє наступній нерівності:

$$F_{k-1} < N < F_k \quad (3.3)$$

При цьому необхідно розрахувати k – максимальне число обчислень цільової функції і довжину відрізка z .

$$z = \frac{b-a}{F_k}; \quad (3.4)$$

3. Визначення значення цільової функції $f_0(u)$ в точці $u = a$ і здійснення першого кроку пошуку, $i=1$.

$$u^1 = a + z \cdot F_{k-2}; \quad (3.5)$$

4. Порівняння значень цільової функції в точках a й u^1 :
якщо $f_0(u^1) > f_0(a)$ (вдалий крок пошуку), то для визначення наступних точок пошуку використовується таке співвідношення:

$$u^{i+1} = u^i + z \cdot F_{k-2-i}, \quad (3.6)$$

після чого здійснюється перехід до пункту 6. Якщо крок пошуку був невдалим ($f_0(u^1) < f_0(a)$), то використовується

$$u^{i+1} = u^i - z \cdot F_{k-2-i}. \quad (3.7)$$

Перехід до пункту 6.

5. Обчислення наступної точки пошуку:

$$u^{i+1} = u^i + z \cdot F_{k-2-i} \cdot \text{sign } \Delta u. \quad (3.8)$$

Функція $\text{sign } \Delta u$ є функцією математичного знаку:

$$\text{sign } \Delta u = \begin{cases} -1, & \text{якщо } u^i - u^{i-1} < 0, \\ 1, & \text{якщо } u^i - u^{i-1} > 0. \end{cases} \quad (3.9)$$

6. Порівняння значень цільової функції в точках відрізка u^i й u^{i+1} .

7. Якщо крок був успішним ($f_0(u^{i+1}) > f_0(u^i)$), тоді необхідно прийняти $i=i+1$ і перейти до пункту 5. Якщо крок пошуку був безуспішним ($f_0(u^{i+1}) < f_0(u^i)$), то тоді із точки u^i здійснюється новий крок:

$$u^{i+2} = u^i - z \cdot F_{k-4-i}. \quad (3.10)$$

Якщо додаткова перевірка значень цільової функції $f_0(u^{i+2})$ й $f_0(u^i)$ вказує на поліпшення значення цільової функції, то $i=i+2$ залишається, і переходимо до пункту 5.

8. Якщо, згідно (3.10), крок пошуку знову є безуспішним, то із точки u^i робиться нова спроба:

$$u^{i+3} = u^i + z \cdot F_{k-4-i}. \quad (3.11)$$

Залишається $i=i+2$ і переходимо до пункту 6.

Алгоритм пошуку, згідно (3.8), (3.10) і (3.11), завершується після того, як оброблено останнє число Фібоначчі. Знайдена при цьому точка u^i з точністю z відповідає екстремальному значенню $f_0(u)$. Наведений алгоритм ілюструється прикладом, наведеним на рис. 3.1. Екстремальне значення знаходиться після чотирьох кроків пошуку і п'яти обчислень значення цільової функції в точці $u=3$.

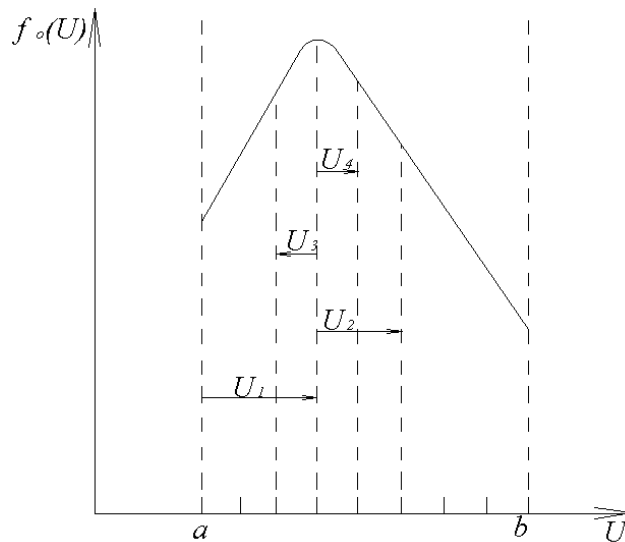


Рис. 3.1. Використання методу Фібоначчі

3.2. Метод золотого перетину.

Недоліком методу Фібоначчі є необхідність попереднього завдання числа експериментів (числа обчислення цільової функції). Однак при розв'язку задачі оптимізації може бути заздалегідь невідоме бажане число експериментів. У цьому випадку використовують так званий метод золотого перетину.

Доведено, що:

$$\lim_{k \rightarrow \infty} \frac{F_{k-1}}{F_k} = \frac{2}{1 + \sqrt{5}} \approx 0,618 \equiv \tau, \quad (3.12)$$

де τ - є розв'язком квадратного рівняння

$$\tau^2 + \tau - 1 = 0. \quad (3.13)$$

Ділення інтервалу невизначеності щоразу здійснюється у відношенні $\frac{\tau}{1-\tau}$. Наприклад, для інтервалу $[0, 1]$ першими пробними внутрішніми точками будуть 0,618 і 0,382. Після цього ліва або права ділянка інтервалу відкидається, залежно від величини $f_0(u)$, а інша знову ділиться у відношенні $\frac{\tau}{1-\tau}$ і т.д.

В методі золотого перетину довжина інтервалу невизначеності на кожному кроці зменшується в τ раз. Можна показати, що для більших k остаточний інтервал у методі золотого перетину всього лише на 17% більше ніж в методі Фібоначчі. При реалізації методу золотого перетину на ЕОМ немає необхідності зберігати в пам'яті або генерувати числа Фібоначчі.

В загальному випадку, якщо права і ліва крайні точки інтервалу невизначеності (позначимо їх через x_L і x_R відповідно) відомі, то координати всіх наступних пробних точок, що отримуються відповідно до методу золотого перетину, можна обчислити за формулами:

$$w = x_R - \tau^n \text{ або } w = x_L + \tau^n,$$

в залежності від того, який з підінтервалів було вилучено на попередньому кроці – лівий чи правий, де n – кількість обчислень значень функції.

Приклад 3.1. Метод золотого перетину

Мінімізувати $f(x) = (100 - x)^2$ в інтервалі $60 \leq x \leq 150$.

Для того, щоб перейти до інтервалу одиничної довжини, проведемо заміну змінної. Нехай $w = (x - 60)/90$, тоді задача приймає наступний вид:

Мінімізувати $f(w) = (40 - 90w)^2$ при обмеженнях $0 \leq w \leq 1$.

Ітерація 1. Інтервал невизначеності $I_1 = (0, 1)$; його довжина $L_1 = 1$. Проведемо два перших обчислення значень функції:

$$w_1 = \tau = 0,618, \quad f(w_1) = 244,0, \\ w_2 = 1 - \tau = \tau^2 = 0,382, \quad f(w_2) = 31,6.$$

Оскільки $f(w_2) < f(w_1)$ і $w_2 < w_1$, інтервал $w \geq w_1$ вилучається.

Ітерація 2. $I_2 = (0, 0,618)$; $L_2 = 0,618 = \tau$. Наступне обчислення значення функції проводиться в точці

$$w_3 = \tau - \tau^2 = \tau(1 - \tau) = \tau^3 = 0,236, \quad f(w_3) = 352.$$

Оскільки $f(w_3) > f(w_2)$ і $w_3 < w_2$, інтервал $w \leq w_3$ вилучається.

Ітерація 3. $I_3 = (0,236, 0,618)$; $L_3 = 0,382 = \tau^2$. Наступні обчислення значення функції проводяться в точці, розміщеній на відстані $\tau \times (\text{довжина інтервалу})$ від лівої крайньої точки інтервалу або на відстані $(1 - \tau) \times (\text{довжина інтервалу})$ від правої крайньої точки. Таким чином,

$$w_4 = 0,618 - (1 - \tau)L_3 = 0,618 - \tau^2 L_3 = 0,618 - \tau^2(\tau^2) = 0,618 - \tau^4 = 0,472; \\ f(w_4) = 6,15.$$

Оскільки $f(w_4) < f(w_2)$ і $w_4 > w_2$, інтервал $w \leq w_2$ вилучається.

В результаті отримано наступний інтервал невизначеності: $0,382 \leq w \leq 0,618$ для змінної w або $94,4 \leq x \leq 115,6$ для змінної x .

Якщо в процесі пошуку проведено шість обчислень значень функції, то довжина результуючого інтервалу для змінної w дорівнює

$$\tau^{N-1} = \tau^5 = 0,09,$$

що відповідає інтервалу довжини 8,1 для змінної x .

3.3. Методи поліноміальної інтерполяції.

Ідея цих методів полягає в апроксимації $f_0(u)$ деякою модельною функцією $f_0^*(u)$, точку мінімуму якої знайти легко, і ця точка береться як поточне наближення шуканої.

Найпростішою модельною функцією буде квадратична:

$$f_0^*(u) = a_0 + a_1(u - u_1) + a_2(u - u_1)(u - u_2). \quad (3.14)$$

Якщо задана послідовність точок u_1, u_2, u_3 і відомі $f_0(u_1), f_0(u_2), f_0(u_3)$, то можна визначити постійні величини a_0, a_1, a_2 такі, що значення $f_0(u)$ й $f_0^*(u)$ співпадуть принаймні в 3-х точках u_1, u_2, u_3 .

При цьому:

$$a_0 = f_0(u_1), a_1 = \frac{f_0(u_2) - f_0(u_1)}{u_2 - u_1}, \quad (3.15)$$

$$a_2 = \frac{1}{u_3 - u_2} \left(\frac{f_0(u_3) - f_0(u_1)}{u_3 - u_1} - \frac{f_0(u_2) - f_0(u_1)}{u_2 - u_1} \right). \quad (3.16)$$

Визначивши точку екстремуму u_4 модельної функції $f_0^*(u)$, необхідно обчислити значення $f_0^*(u_4)$ й $f_0(u_4)$. У тому випадку, якщо ці значення будуть досить близькі, можна вважати, що знайдено екстремум $f_0(u)$. Якщо значення $f_0(u_4)$ й $f_0^*(u_4)$ будуть відрізнятися, то необхідно повторити процедуру квадратичної інтерполяції, відкинувши величини u_1 й $f_0(u_1)$ і прийнявши, що $u_1 = u_2$, $f_0(u_1) = f_0(u_2)$, $u_2 = u_3$, $f_0(u_2) = f_0(u_3)$, $u_3 = u_4$, $f_0(u_3) = f_0(u_4)$. Аналогічні операції необхідно проводити доти, поки не буде досягнутий бажаний збіг $f_0(u)$ й $f_0^*(u)$ у допустимій точці екстремуму.

Необхідно відзначити, що якщо є чотири числові характеристики $u_1 \div u_4$ і $f_0(u_1) \div f_0(u_4)$, то можна здійснювати кубічну інтерполяцію функції $f_0(u)$.

3.4. Метод послідовного оцінювання із використанням квадратичної апроксимації.

Цей метод, розроблений Пауелом, заснований на послідовному застосуванні процедури оцінювання з використанням квадратичної апроксимації. Схему алгоритму можна описати в такий спосіб. Нехай x_1 — початкова точка, Δx — обрана величина кроку по осі x .

Крок 1. Обчислити $x_2 = x_1 + \Delta x$.

Крок 2. Обчислити $f(x_1)$ і $f(x_2)$.

Крок 3. Якщо $f(x_1) > f(x_2)$, припускаємо, що $x_3 = x_1 + 2\Delta x$. Якщо $f(x_1) \leq f(x_2)$, припускаємо $x_3 = x_1 - \Delta x$.

Крок 4. Обчислити $f(x_3)$ і знайти: $F_{\min} = \min\{f_1, f_2, f_3\}$, $X_{\min} = \text{точка } x_i, \text{ відповідна } F_{\min}$

Крок 5. По трьох точках x_1, x_2, x_3 обчислити \bar{x} , використовуючи формулу для оцінювання за допомогою квадратичної апроксимації.

Крок 6. Перевірка на закінчення пошуку.

а) чи є різниця $F_{\min} - f(\bar{x})$ досить малою?

б) чи є різниця $X_{\min} - \bar{x}$ досить малою?

Якщо обидві умови виконуються, пошук закінчується. В іншому випадку перейти до кроку 7.

Крок 7. Вибрати «найкращу» точку (X_{\min} або \bar{x}) і дві точки з обох сторін від неї. Позначити ці точки в природному порядку і перейти до кроку 4.

При першій реалізації кроку 5 границі інтервалу, що містить точку мінімуму, необов'язково виявляються встановленими. При цьому отримана точка \bar{x} може перебувати за точкою x_3 . Для того щоб виключити можливість занадто великого екстраполяційного переміщення, варто провести після кроку 5 додаткову перевірку і у випадку, коли точка \bar{x} знаходиться занадто далеко від x_3 , замінити \bar{x} точкою, координата якої обчислюється із врахуванням заздалегідь установленної довжини кроку.

Приклад 3.2. Метод Пауела

Розглянемо задачу із прикладу 1: мінімізувати $f(x) = 2x^2 + (16/x)$. Нехай початкова точка x_1 і довжина кроку Δx . Для перевірки на закінчення пошуку використовуються наступні параметри збіжності:

$$\left| \frac{\text{Різниця значень } x}{x} \right| \leq 3 \times 10^{-2}, \left| \frac{\text{Різниця значень } f}{f} \right| \leq 3 \times 10^{-3}.$$

Ітерація 1

Крок 1. $x_2 = x_1 + \Delta x = 2$.

Крок 2. $f(x_1) = 18, f(x_2) = 16$.

Крок 3. $f(x_1) > f(x_2)$, отже приймаємо $x_3 = 1 + 2 = 3$.

Крок 4. $f(x_3) = 23.33, F_{\min} = 16, X_{\min} = x_2$.

Крок 5. $a_1 = \frac{16-18}{2-1} = -2$,

$$a_2 = \frac{1}{3-2} \left\{ \frac{23.33-18}{3-1} - a_1 \right\} = \frac{5.33}{2} + 2 = 4.665,$$

$$\bar{x} = \frac{1+2}{2} - \frac{(-2)}{2(4.665)} = 1.5 + \frac{1}{4.665} = 1.714,$$

$$f(x) = 15,210.$$

Крок 6. Перевірка на закінчення пошуку:

$$a) \left| \frac{16 - 15,210}{15,210} \right| = 0,0519 > 0,003.$$

Отже, продовжуємо пошук.

Крок 7. Вибираємо x як «найкращу» точку, а $x_1=1$ і $x_2=2$ – як точки, які її оточують. Позначаємо ці точки в природному порядку і переходимо до ітерації 2, що починається із кроку 4.

Ітерація 2

Крок 4. $x_1=1$, $f_1=18$, $x_2=1,714$, $f_2=15,210=F_{\min}$, $X_{\min}=x_2$, $x_3=2$, $f_3=16$.

$$\text{Крок 5. } a_1 = \frac{15,210 - 18}{1,714 - 1} = -3,908,$$

$$a_2 = \frac{1}{2 - 1,714} \left\{ \frac{16 - 18}{2 - 1} - (-3,908) \right\} = \frac{1,908}{0,286} = 6,671,$$

$$\bar{x} = \frac{2,714}{2} - \frac{(-3,908)}{2(6,671)} = 1,6125,$$

$$f(\bar{x}) = 15,123.$$

Крок 6. Перевірка на закінчення пошуку:

$$a) \left| \frac{15,210 - 15,142}{15,142} \right| = 0,0045 > 0,003 \text{ (умова не виконується)}$$

Крок 7. Вибираємо x як «найкращу» точку, а $x_1=1$ й $x_2=1,714$ – як точки, які її оточують.

Ітерація 3

Крок 4. $x_1=1$; $f_1=18$; $x_2=1,65$; $f_2=15,142=F_{\min}$; $X_{\min}=x_2$; $x_3=1,714$; $f_3=15,210$.

$$\text{Крок 5. } a_1 = \frac{15,142 - 18}{1,65 - 1} = -4,397,$$

$$a_2 = \frac{1}{1,714 - 1,650} \left\{ \frac{15,210 - 18}{1,714 - 1} - (-4,397) \right\} = 7,647,$$

$$\bar{x} = \frac{2,65}{2} - \frac{(-4,397)}{2(7,647)} = 1,6125,$$

$$f(\bar{x}) = 15,123.$$

Крок 6. Перевірка на закінчення пошуку:

$$a) \left| \frac{15,142 - 15,123}{15,123} \right| = 0,0013 < 0,003,$$

$$б) \left| \frac{1,65 - 1,6125}{1,6125} \right| = 0,023 < 0,03.$$

Отже, пошук закінчено.

3.5. Метод пошуку з використанням кубічної апроксимації

Відповідно до розглянутого методу підлягаючої мінімізації функція $f(\bar{x})$ апроксимується поліномом третього порядку. Логічна схема методу аналогічна схемі методу з використанням квадратичної апроксимації. Однак у цьому випадку побудова апроксимуючого полінома проводиться на основі меншого числа точок, оскільки в кожній точці можна обчислювати значення як функції, так і її похідної.

Робота алгоритму починається в довільно обраній точці x_1 : знаходиться інша точка x_2 , така, що похідні $f'(x_1)$ і $f'(x_2)$ мають різні знаки. Інакше кажучи, необхідно вмістити стаціонарну точку \bar{x} , в якій $f'(\bar{x}) = 0$, в інтервал між x_1 і x_2 . Апроксимуюча кубічна функція записується в наступному вигляді:

$$\bar{f}(x) = a_0 + a_1(x - x_1) + a_2(x - x_1)(x - x_2) + a_3(x - x_1)^2(x - x_2) \quad (3.17)$$

Параметри рівняння (3.17) підбираються таким чином, щоб значення $\bar{f}(x)$ і її похідної у точках x_1 і x_2 збігалися зі значеннями $f(x)$ і $f'(x)$ в цих точках. Перша похідна функції $\bar{f}(x)$ дорівнює:

$$\frac{d\bar{f}(x)}{dx} = a_1 + a_2(x - x_1) + a_2(x - x_2) + a_3(x - x_1)^2 + 2a_3(x - x_1)(x - x_2) \quad (3.18)$$

Коефіцієнти a_0 , a_1 , a_2 й a_3 рівняння (1) визначаються за відомим значенням $f(x_1)$, $f(x_2)$, $f'(x_1)$ і $f'(x_2)$ шляхом розв'язку наступної системи лінійних рівнянь:

$$f_1 = f(x_1) = a_0, \quad (3.19)$$

$$f_2 = f(x_2) = a_0 + a_1(x_2 - x_1), \quad (3.20)$$

$$f'_1 = f'(x_1) = a_1 + a_2(x_1 - x_2), \quad (3.21)$$

$$f'_2 = f'(x_2) = a_1 + a_2(x_1 - x_2) + a_3(x - x_1)^2. \quad (3.22)$$

Помітимо, що дана система легко вирішується рекурсивним методом. Після того як коефіцієнти знайдені, діючи за аналогією з випадком квадратичної апроксимації, можна оцінити координату стаціонарної точки функції за допомогою апроксимуючого полінома (3.17). При цьому прирівнювання до нуля похідної (3.18) приводить до квадратного рівняння. Використовуючи формулу для обчислення коренів квадратного рівняння, запишемо розв'язок, що визначає стаціонарну точку апроксимуючого кубічного полінома, у наступному вигляді:

$$\bar{x} = \begin{cases} x_2, & \text{якщо } \mu < 0, \\ x_2 - \mu(x_2 - x_1), & \text{якщо } 0 \leq \mu \leq 1, \\ x_1, & \text{якщо } \mu > 1, \end{cases} \quad (3.23)$$

де:

$$\mu = \frac{f_2' + \omega - z}{f_2' - f_1' + 2\omega}, \quad (3.24)$$

$$z = \left(\frac{3(f_1 - f_2)}{x_2 - x_1} \right) + f_1' + f_2', \quad (3.25)$$

$$\omega = \begin{cases} (z^2 - f_1' f_2')^{1/2}, & \text{якщо } x_1 < x_2, \\ -(z^2 - f_1' f_2')^{1/2}, & \text{якщо } x_1 > x_2. \end{cases} \quad (3.26)$$

Формула (3.26) забезпечує належний вибір одного із двох коренів квадратного рівняння; для значень μ , що знаходяться в інтервалі від 0 до 1, формула (3.23) гарантує, що отримувана точка \bar{x} розташована між x_1 і x_2 . Потім знову вибираються дві точки для реалізації процедури кубічної апроксимації — \bar{x} і одна із точок x_1 або x_2 , причому значення похідної досліджуваної функції в цих точках повинні бути протилежні за знаком, і процедура кубічної апроксимації повторюється.

Наведемо формалізований опис алгоритму. Нехай задані початкова точка x_0 , позитивна величина кроку Δ і параметри збіжності ε_1 й ε_2 .

Крок 1. Обчислити $f'(x_0)$.

Якщо $f'(x_0) < 0$, то обчислити $x_{k+1} = x_k + 2^k \Delta$ для значень $K=0, 1, \dots$

Якщо $f'(x_0) > 0$, то обчислити $x_{k+1} = x_k - 2^k \Delta$ для значень $K=0, 1, \dots$

Крок 2. Обчислити значення $f'(x)$ в точках x_{k+1} при $K=0, 1, 2, \dots$ аж до точки x_M , у якій $f'(x_{M-1})f'(x_M) \leq 0$. Потім прийняти $x_1 = x_{M-1}$, $x_2 = x_M$. Обчислити значення f_1, f_2, f_1' і f_2' .

Крок 3. Знайти стаціонарну точку \bar{x} апроксимуючого кубічного поліному, користуючись формулою (2.10).

Крок 4. Якщо $f(x) < f(x_1)$, то перейти до кроку 5. У протилежному випадку обчислювати \bar{x} по формулі $\bar{x} = \bar{x} + \frac{1}{2}(\bar{x} - x)$ доти, поки не буде виконуватися нерівність $f(\bar{x}) \leq f(x_1)$.

Крок 5. Перевірка на закінчення пошуку.

Якщо $|f'(\bar{x})| \leq \varepsilon$ і $|(\bar{x} - x_1/\bar{x})| \leq \varepsilon_2$, пошук закінчити. Інакше прийняти або

а) $x_2 = x_1$ і $x_1 = \bar{x}$, якщо $f'(\bar{x})f'(x_1) < 0$, або

б) $x_1 = \bar{x}$, якщо $f'(\bar{x})f'(x_2) < 0$.

Потім перейти до кроку 3.

Помітимо, що кроки 1 і 2 реалізують процедуру пошуку меж інтервалу евристичним методом, до того ж зміна знаку похідної використовується як критерій переходу через точку оптимуму. На кроці 3 проводяться обчислення координати точки оптимуму апроксимуючого

поліному. Крок 4 асоційований з перевіркою того факту, що отримана оцінка дійсно є поліпшеним наближенням до точки оптимуму. У випадку, коли значення похідної обчислюються безпосередньо, метод пошуку з використанням кубічної апроксимації, безумовно, є більш ефективним у порівнянні з кожним із наведених вище методів пошуку. Однак якщо значення похідної обчислюються шляхом диференціювання різниць, то перевагу варто віддати методу Пауела, який базується на квадратичній апроксимації.

Приклад 3.2. Пошук із використанням кубічної апроксимації і похідних

Знову розглянемо задачу, у якій потрібно мінімізувати $f(x) = 2x^2 + \frac{16}{x}$. Нехай початкова точка $x_0 = 1$ і довжина кроку $\Delta = 1$. Як параметри збіжності використаємо: $\varepsilon_1 = 10^{-2}$ і $\varepsilon_2 = 3 \times 10^{-2}$,

$$f'(x) = \frac{df}{dx} = 4x - \frac{16}{x^2}.$$

Ітерація 1

Крок 1. $f'(1) = -12 < 0$. Отже, $x_1 = 1 + 1 = 2$.

Крок 2. $f'(2) = 4$.

Тому що $f'(1)f'(2) = -48 < 0$, стаціонарна точка розташована між 1 й 2. Прийmemo, $x_1 = 1$, $x_2 = 2$. Тоді $f_1 = 18$, $f_2 = 16$, $f'_1 = -12$, $f'_2 = 4$.

Крок 3.

$$z = 3/1(18 - 16) + (-12) + 4 = -2,$$

$$\omega = +[4 - (-12)(4)]^{1/2} = (52)^{1/2} = 7,211,$$

$$\mu = \frac{4 + 7,211 - (-2)}{4 - (-12) + 2(7,211)} = 0,4343,$$

$$\bar{x} = 2 - 0,4343(2 - 1) = 1,5657.$$

Крок 4. $f(1,5657) = 15,1219 < f(x_1) = 18$. Отже, продовжуємо пошук.

Крок 5. Перевірка на закінчення пошуку.

$f'(1,5657) = -0,2640$; пошук не закінчений. Тому що $f'(\bar{x})f'(x_2) = (-0,2640)(4) < 0$,

приймаємо $x_1 = \bar{x} = 1,5657$.

Ітерація 2

Крок 3.

$$z = \frac{3}{0,4343}(15,1219 - 16) + (-0,2640) + 4 = -2,3296,$$

$$\omega = +[(2,3296)^2 - (-0,2640)(4)]^{1/2} = 2,5462,$$

$$\mu = \frac{4 + 2,5462 - (-2,3296)}{4 - (-0,2640) + 2(2,5462)} = 0,9486,$$

$$\bar{x} = 2 - 0,9486(2 - 1,5657) = 1,5880.$$

Крок 4. $f(1,5880) = 15,1191$; $f(x_1) = 15,1219$. Отже, продовжуємо пошук.

Крок 5. Перевірка на закінчення пошуку: $f'(1,5880) = 0,0072 < 10^{-2}$,

$$\left| \frac{1,5880 - 1,5657}{1,5880} \right| = 0,0140 < 3 \times 10^{-2}.$$

Пошук закінчено.

3.6. Застосування методів одномірної оптимізації в задачах ідентифікації об'єктів управління.

При побудові адаптивних систем автоматичного регулювання промисловими об'єктами актуальною задачею є оцінка параметрів об'єкту з метою визначення початкових значень коефіцієнта передачі регулятора $K_p^{(0)}$ і часу інтегрування $T_n^{(0)}$ при введенні в дію регуляторів чи за ініціативним сигналом оператора, а для нелінійних об'єктів – і при зміні навантаження.

Ідентифікація об'єкта, що описується лінійними диференціальними рівняннями зі сталими коефіцієнтами та запізненням, може бути виконана за результатами аналізу початкової ділянки перехідної характеристики шляхом визначення моменту t_1 досягнення максимуму першої похідної вихідної величини $y(t)$ і значення вихідної величини $y(t)$ та її першої похідної $\dot{y}(t)$ в цей момент часу. Процес ідентифікації починається з подачі на вхід об'єкта в розімкненому контурі зі сторони регулюючого органа ступінчатого збурення. Вихідний сигнал об'єкта подається через аперіодичну ланку зі сталою часу T_ϕ на аналізатор початкової ділянки кривої, в якому за допомогою процедури одномірного пошуку екстремуму визначається час t_1 , а також обчислюються значення $y(t_1)$ і $\dot{y}(t_1)$.

Експериментально встановлено, що зі зміною сталої часу об'єкта однозначно змінюється час t_1 , а коефіцієнт підсилення об'єкта визначається через час t_1 і значення $\dot{y}(t_1)$. Загальний вид апроксимуючої функції для залежності $T_{об} = f(t_1)$ має вигляд:

$$T_{об} = k_1 + k_2(t_1 + k_3)^{k_4}, \quad (3.27)$$

де k_i – сталі коефіцієнти, $i = 1 \div 4$, конкретні значення яких обчислюються окремо для різних значень T_ϕ , що вибираються експериментально, виходячи з діапазону значень $T_{об}$, для якого будується апроксимуюча залежність.

Для визначення $K_{об}$ було отримано наступну формулу:

$$K_{OB} = |\dot{y}(t)| \cdot (K_1 + K_2 \cdot (t_1 + K_3)^{K_4}), \quad (3.28)$$

де K_i – сталі коефіцієнти, $i = 1 \div 4$.

Наприклад, для $T_\phi = 100$ отримано наступні розрахункові формули:

$$T_{OB} = -23.385 + 1.461 \cdot 10^{-13} (t_1 + 257.66)^{5.841};$$

$$K_{OB} = |\dot{y}(t_1)| \cdot (79.198 + 1.56 \cdot 10^{-4} \cdot (t_1 + 86.506)^{2.686}).$$

При цьому експериментальним шляхом було встановлено, що різниця між реальними і розрахунковими параметрами об'єкта незначна.

3.7. Застосування методів одномірної оптимізації в задачах адаптивного управління.

Рівняння адаптивного ПІ-регулятора має вигляд:

$$u(t) = K_p^{(n+1)} \left(\varepsilon(t) + \frac{1}{T_u} \int \varepsilon(t) dt + T_\phi \dot{\varepsilon}(t) \right); \quad (1.29)$$

$$K_p^{n+1} = K_p^n \cdot \alpha \cdot A; \quad T_\phi = \beta \cdot T_u; \quad A = \frac{|\varepsilon(t)|}{(t_1 |\dot{\varepsilon}(t)|)}.$$

α, β – сталі коефіцієнти.

Для визначення динамічних і статичних властивостей об'єкта алгоритм адаптації використовує інформацію про час t_1 досягнення максимуму модуля першої похідної вихідної величини $y(t)$ (або сигналу розбалансу $\varepsilon(t)$) в точці перегину (рис. 3.2), а також про модуль $y(t_1)$ похідної від $y(t_1)$ при поданні тестового збурення за завданням або за вихідною величиною об'єкта.

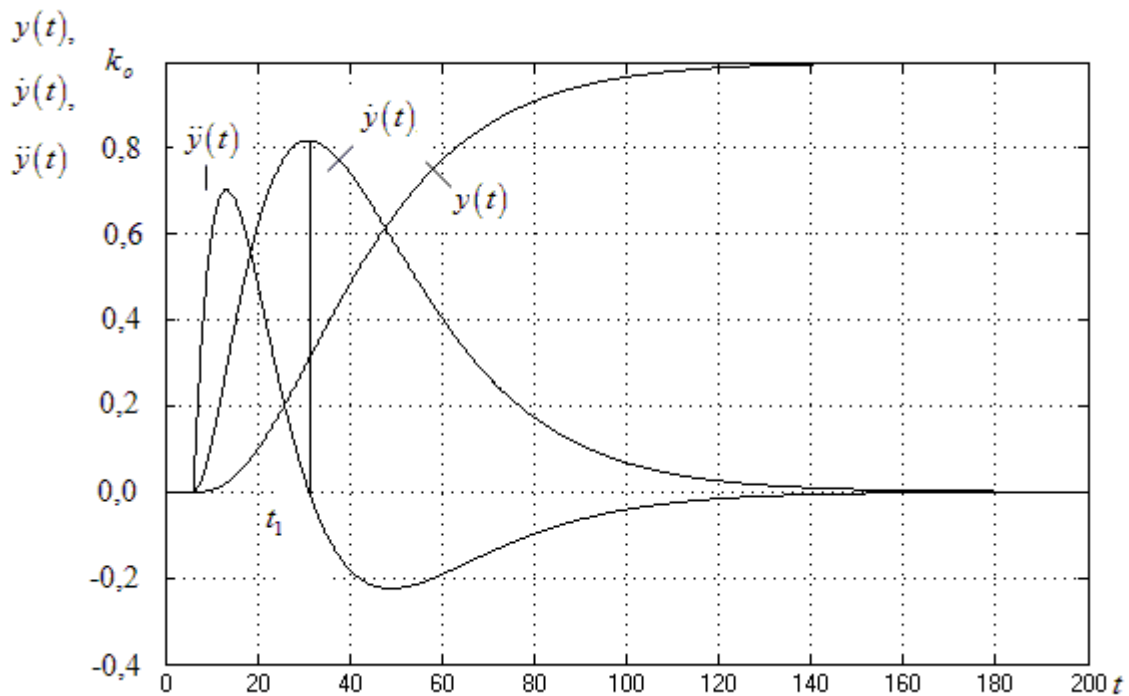


Рис. 3.2. *Перехідна характеристика, похідна і друга похідна від неї.*

Покроковий алгоритм знаходження коефіцієнтів рівняння (1.29) представлено нижче.

Крок 1. Подача тестового збурення.

Крок 2. Визначення максимуму першої похідної від сигналу розбалансу. Для цього використовується алгоритм порівняння двох останніх вимірів модуля першої похідної сигналу розбалансу $|\dot{\varepsilon}(t_i)|$ та $|\dot{\varepsilon}(t_{i-1})|$. В той момент часу, коли нове значення модуля похідної сигналу розбалансу $|\dot{\varepsilon}(t_i)|$ стає меншим за попереднє $|\dot{\varepsilon}(t_{i-1})|$, пошук максимуму завершується. При цьому за максимальне значення приймається $|\dot{\varepsilon}(t_1)| = |\dot{\varepsilon}(t_{i-1})|$, момент досягнення цього максимуму – $t_1 = t_{i-1}$. Крім того, визначається значення модуля сигналу розбалансу в момент t_1 – $|\varepsilon(t_1)|$.

Крок 3. За отриманими значеннями $|\dot{\varepsilon}(t_1)|$, $|\varepsilon(t_1)|$ і t_1 виконується підстроювання коефіцієнта підсилення регулятора за формулою:

$$K_p^{(n+1)} = K_p^{(n)} \cdot \alpha \cdot \frac{|\varepsilon(t_1)|}{|\dot{\varepsilon}(t_1)| \cdot t_1}, \quad (1.30)$$

де n – номер кроку адаптації, α – емпіричний коефіцієнт, що задає якість перехідного процесу. Діапазон зміни α 1...1,8. Рекомендоване значення $\alpha = 1,44$.

На першій ітерації адаптації використовується початкове значення коефіцієнта підсилення регулятора $K_p^{(0)}$, що може бути визначено за допомогою інженерних методів.

Крок 4. Порівняння нового значення коефіцієнта підсилення регулятора $K_p^{(n+1)}$ зі старим $K_p^{(n)}$. Якщо модуль їх різниці менше наперед заданого значення Δ , то процес підстроювання завершується і за оптимальне значення коефіцієнта підсилення приймається його останнє значення – $K_p^{(n+1)}$, в противному випадку необхідно повторити кроки 1–4 при новому значенні коефіцієнта підсилення $K_p^{(n+1)}$. Рекомендоване значення $\Delta = 0,1$. Для досягнення більш високих показників якості необхідно задати менше значення Δ .

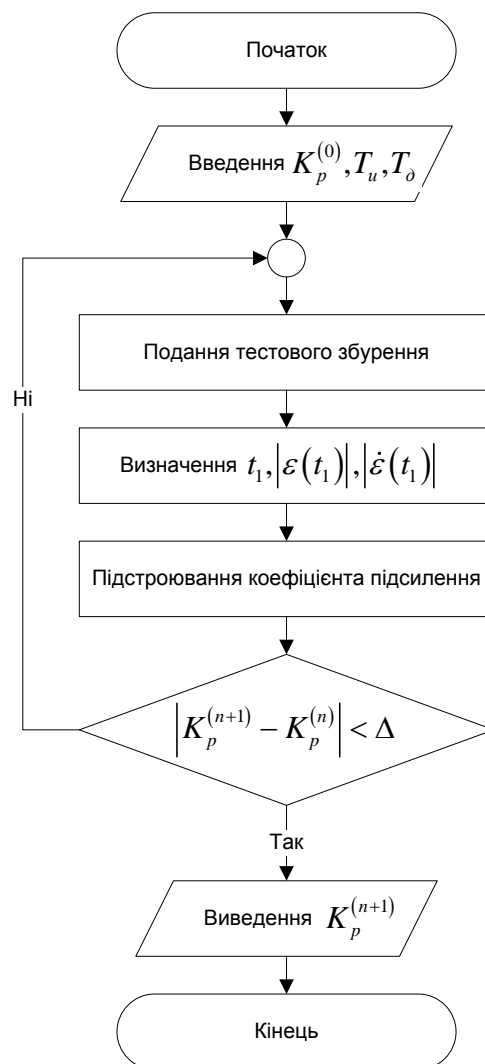


Рис. 3.3. Структурна схема алгоритму корекції коефіцієнта підсилення

3.8. Контрольні завдання

1. Реалізуйте процедуру одномірного пошуку точки оптимуму функції

$$f(x) = 3x^2 + \frac{12}{x^3} - 5 \text{ в інтервалі } \frac{1}{2} \leq x \leq \frac{5}{2},$$

використовуючи

а) метод золотого перетину,

б) метод Пауела,

в) метод пошуку з використанням кубічної апроксимації і похідних.

В кожному випадку проведіть по чотири обчислення значення функції. Порівняйте результуючі інтервали пошуку, отримані за допомогою перелічених вище методів.

2. Якщо $F_0 = 1$ і $F_1 = 1$, а $F_n = F_{n-1} + F_{n-2}$ для $n \geq 2$, то покажіть, що

$$F_n = \frac{\tau^{n+1} - (-\tau)^{-(n+1)}}{\sqrt{5}},$$

$$\text{де } \tau = \frac{1 + \sqrt{5}}{2} \approx 1,618033989.$$

Використовуючи це рекурентне співвідношення, покажіть, що $F_2 = 2$, $F_3 = 3$, $F_5 = 8$, $F_6 = 13$, $F_{10} = 89$, $F_{20} = 10946$.

3. Покажіть, що

$$а) F_n \approx \frac{\tau^{n+1}}{\sqrt{5}} \text{ для великих } n;$$

$$б) \frac{F_{n-1}}{F_n} \approx \frac{1}{\tau} \text{ для великих } n.$$

4. Покажіть, що для скорочення інтервалу невизначеності на 1% від початкової величини, потрібно виконати 11 обчислень функції при використанні методу Фібоначчі. Якщо розміщення точок задати на початку, яким буде мінімальне число необхідних точок?

5. Застосуйте метод Фібоначчі за наявності 10 обчислень функції для визначення мінімуму функції $f(x) = 2x^2 + 3e^{-x}$ на інтервалі $[0;1]$.

6. За допомогою метода золотого перетину визначте мінімум функції $2x^2 + 3e^{-x}$ з точністю до двох десяткових знаків. Початковий інтервал

невизначеності $[0;1]$. Скільки разів необхідно обчислити функцію? Порівняйте з завданням 5.

7. Застосуйте квадратичну інтерполяцію для визначення точки мінімуму функції $f(x) = -e^{-x} \ln(x)$ на інтервалі $[1;3]$ з точністю 0,001.

4. ГРАДІЄНТНІ МЕТОДИ БАГАТОМІРНОЇ ОПТИМІЗАЦІЇ

4.1. Методи першого порядку

4.1.1. Градієнтний метод

Як відомо (див. розділ 1.5), градієнт (антиградієнт) скалярної функції $f_0(\vec{u})$ в деякій точці \vec{u}_k спрямований у бік найшвидшого зростання (найшвидшого зменшення) функції.

Вибираючи як напрямок пошуку екстремуму градієнт або антиградієнт (залежно від постановки задачі), алгоритм градієнтного методу можна записати у вигляді:

$$\vec{u}_{k+1} = \vec{u}_k - h_k \cdot \nabla f_0(\vec{u}_k), \quad (4.1)$$

де h_k - кроковий множник (параметр кроку).

Останній запис означає, що із точки $\vec{u}_k = \{u_1, u_2, \dots, u_n\} \big|_{\vec{u}=\vec{u}_k}$ у напрямку екстремуму робиться пошуковий крок величиною: $\Delta \vec{u}_k = \left\{ h \frac{\partial f_0(\vec{u})}{\partial u_1}, h \frac{\partial f_0(\vec{u})}{\partial u_2}, \dots, h \frac{\partial f_0(\vec{u})}{\partial u_n} \right\} \big|_{\vec{u}=\vec{u}_k}$ по кожній із незалежних змінних $u_i, i=1, 2, \dots, n$, тобто одночасно змінюються значення всіх незалежних змінних (рис. 4.1). У знайденій точці \vec{u}_{k+1} обчислюється значення цільової функції $f_0(\vec{u}_{k+1})$ і виконується перевірка достатніх умов оптимальності.

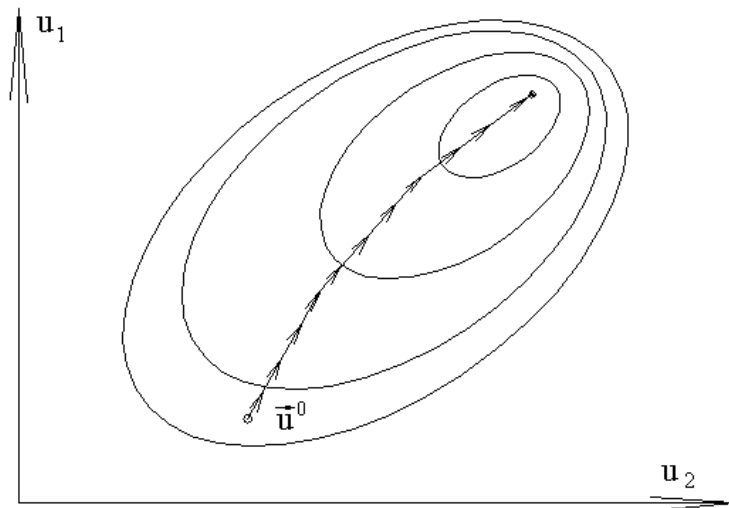


Рис. 4.1. Пошук екстремуму градієнтним методом

На практиці, як правило, пошук припиняють при виконанні умови:

$$\|\nabla f_0(\vec{u})\|_{\vec{u}=\vec{u}_k} = \sqrt{\sum_{i=1}^n \left[\left(\frac{\partial f_0(\vec{u})}{\partial u_i} \right) \Big|_{\vec{u}=\vec{u}_k} \right]^2} \leq \varepsilon,$$

де символ $\| \cdot \|$ позначає норму градієнта цільової функції $f_0(\vec{u})$; ε — наперед задане, достатньо мале число.

Як видно з (4.1) величина пошукового кроку залежить від градієнта цільової функції $\nabla f_0(\vec{u}_k)$. Це обставина істотно впливає на процес пошуку, тому що поблизу екстремуму $\nabla f_0(\vec{u}_k) \rightarrow 0$, і, отже, зменшується величина пробного кроку, що зменшує швидкість збіжності (швидкість досягнення екстремуму) алгоритму (4.1) до екстремуму.

Прийнятна величина пробного кроку встановлюється за допомогою крокового множника h_k . Необхідно відзначити, що занадто мале значення h_k знижує швидкість збіжності, а занадто велике значення h_k може призвести до того, що екстремум не буде досягнутий взагалі. Останнє пояснюється тим, що при великому h_k відбувається перехід точки \vec{u}_{k+1} через точку екстремуму, і надалі зміну напрямку пошуку практично на протилежний, тобто повернення в точку \vec{u}_k . Таким чином, у районі екстремуму виникають пошукові коливання, що не припиняються.

Для вибору величини крокового множника може бути використаний наступний алгоритм:

$$h_{k+1} = \begin{cases} 2h_k, & \text{якщо } \gamma_k < \gamma_{\min}, \\ h_k, & \text{якщо } \gamma_{\min} \leq \gamma_k \leq \gamma_{\max}, \\ (\frac{1}{3} \div \frac{2}{5})h_k, & \text{якщо } \gamma_k > \gamma_{\max}, \end{cases} \quad (4.2)$$

де γ_k - кут між сусідніми градієнтами (рис. 4.2), а γ_{\min} й γ_{\max} — допустимі граничні значення.

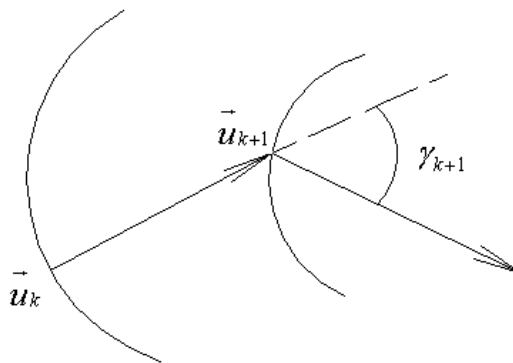


Рис. 4.2. До алгоритму вибору кроку градієнтного метода

Кут γ_{k+1} може бути обчислений в такий спосіб:

$$\cos \gamma_{k+1} = \frac{\sum_{i=1}^m \left(\frac{\partial f_0(\vec{u})}{\partial u_i} \Big|_{\vec{u}=\vec{u}_k} \cdot \frac{\partial f_0(\vec{u})}{\partial u_i} \Big|_{\vec{u}_k} \right)}{\sqrt{\sum_{i=1}^m \left(\frac{\partial f_0(\vec{u})}{\partial u_i} \Big|_{\vec{u}_{k+1}} \right)^2 \cdot \sum_{i=1}^m \left(\frac{\partial f_0(\vec{u})}{\partial u_i} \Big|_{\vec{u}_k} \right)^2}} \quad (4.3)$$

З рис. 4.2 випливає, що якщо γ_{k+1} по величині малий, тобто напрямки $\nabla f_0(\vec{u}_k)$ і $\nabla f_0(\vec{u}_{k+1})$ практично збігаються, то можна збільшити пошуковий крок, припускаючи, що екстремум перебуває далеко. Якщо ж γ_{k+1} — великий, тобто градієнти $\nabla f_0(\vec{u}_k)$ і $\nabla f_0(\vec{u}_{k+1})$ спрямовані практично назустріч один одному, то необхідно зменшити пошуковий крок, тому що очевидно, що точка \vec{u}_{k+1} перейшла за точку екстремуму. Ці прості міркування і покладені в основу алгоритму (4.1).

4.1.2. Метод найшвидшого спуску.

Градiєнтний метод (4.1), на кожній ітерації якого величина крокового множника h_k вибирається з умови мінімуму функції $f_0(\vec{u})$ в напрямку руху, тобто $f_0(\vec{u}_k - h_k \cdot \nabla f_0(\vec{u}_k)) = \min_{h \geq 0} f_0(\vec{u}_k - h \cdot \nabla f_0(\vec{u}_k))$, називається *методом найшвидшого спуску (методом Коші)*.

Відповідно до методу найшвидшого спуску, на кожній ітерації необхідно за допомогою описаних вище методів вирішувати задачу одновимірної мінімізації функції $f_0(\vec{u})$ в напрямку антиградієнту. Пошук здійснюється доти, поки не перестане зменшуватися значення $f_0(\vec{u})$ (рис. 4.3). Після цього визначається значення $\nabla f_0(\vec{u}_{k+1})$ і процедура пошуку триває. Якщо $\|\nabla f_0(\vec{u}_{k+1})\| \leq \varepsilon$, то пошук припиняється.

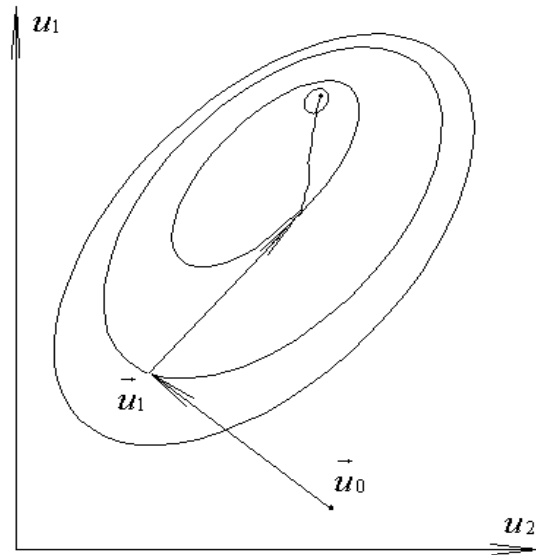


Рис. 4.3. Траєкторія руху за методом Коші

Особливістю методу найшвидшого спуску є те, що для квадратичних цільових функцій, напрямки спуску на двох послідовних ітераціях взаємно ортогональні (перпендикулярні).

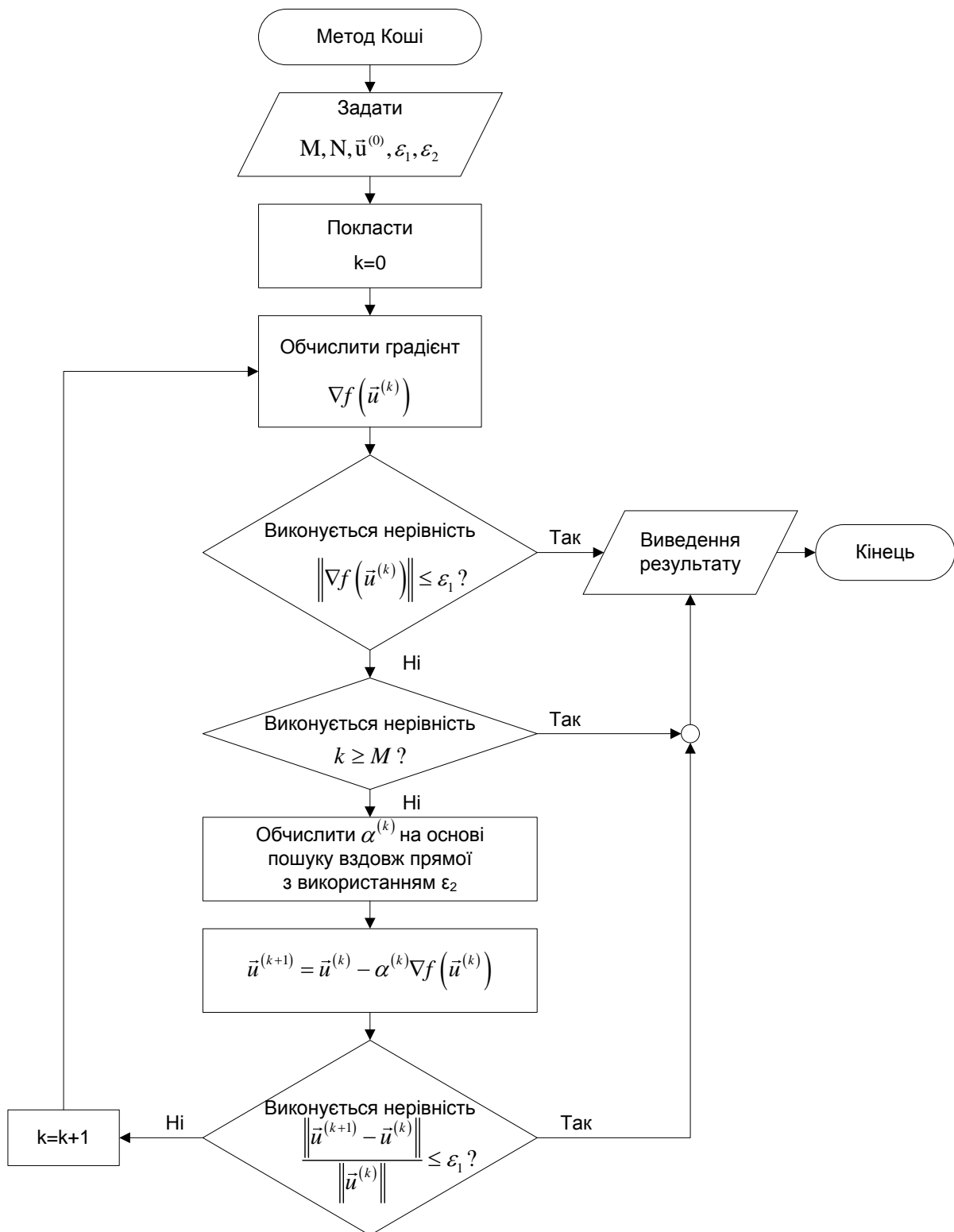


Рис. 4.4. Структурна схема методу Коші

Приклад 4.1. Метод найшвидшого спуску

Розглянемо функцію

$$f(x) = 8x_1^2 + 4x_1x_2 + 5x_2^2$$

і скористаємось методом Коші для вирішення задачі її мінімізації.

Перш за все, обчислимо компоненти градієнта

$$\frac{\partial f}{\partial x_1} = 16x_1 + 4x_2, \quad \frac{\partial f}{\partial x_2} = 10x_2 + 4x_1.$$

Для того, щоб застосувати метод найшвидшого спуску, задамо початкове наближення

$$x^{(0)} = [10, 10]^T$$

і побудуємо нове наближення

$$x^{(1)} = x^{(0)} - \alpha^{(0)} \nabla f(x^{(0)}).$$

Оберемо $\alpha^{(0)}$ таким чином, щоб $f(x^{(1)}) \rightarrow \min$; $\alpha^{(0)} = 0,056$. Отже, $x^{(1)} = [-1, 20, 2, 16]^T$. Далі знайдемо точку

$$x^{(2)} = x^{(1)} - \alpha^{(1)} \nabla f(x^{(1)}),$$

обчисливши градієнт в точці $x^{(1)}$ і провівши пошук вздовж прямої.

Таблиця 4.1

Результати обчислення за методом Коші

k	$x_1^{(k)}$	$x_2^{(k)}$	$f(x^{(k)})$
1	-1,2403	2,1181	24,2300
2	0,1441	0,1447	0,3540
3	-0,0181	0,0309	0,0052
4	0,0021	0,0021	0,0000

В табл. 4.1 представлено дані, отримані при проведенні ітерацій на основі одномірного пошуку за методом квадратичної інтерполяції. Послідовність отриманих точок зображено на рис. 4.5.

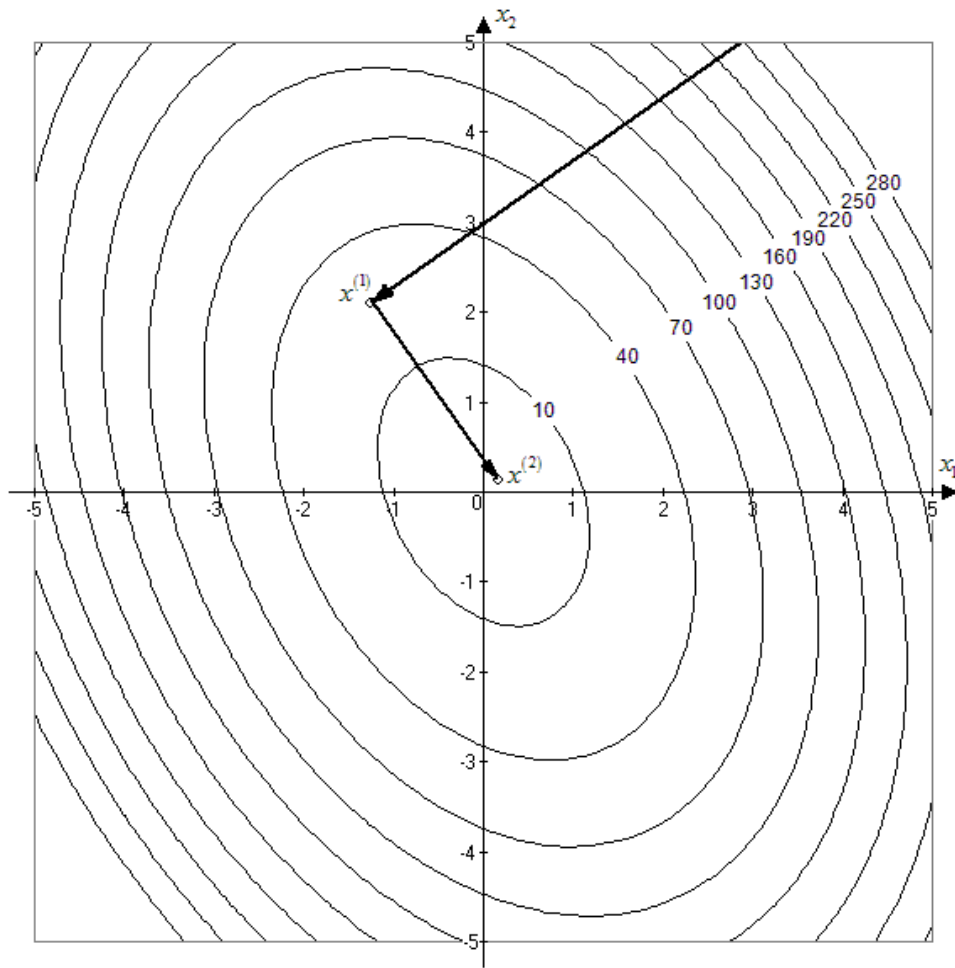


Рис. 4.5. Ітерації за методом Коші з використанням квадратичної інтерполяції

4.1.3. Метод крутого спуску

В методі крутого спуску в напрямку градієнта виконується не один крок, а декілька до моменту досягнення локального екстремуму.

Алгоритм метода має вигляд

$$\vec{u}_{k+1} = \vec{u}_k - h_k \cdot \frac{df(x^{(p)})}{dx_j} \quad (4.4)$$

де p – точка, в якій обчислюється градієнт.

Траєкторію руху за цим алгоритмом зображено на рис. 4.6.

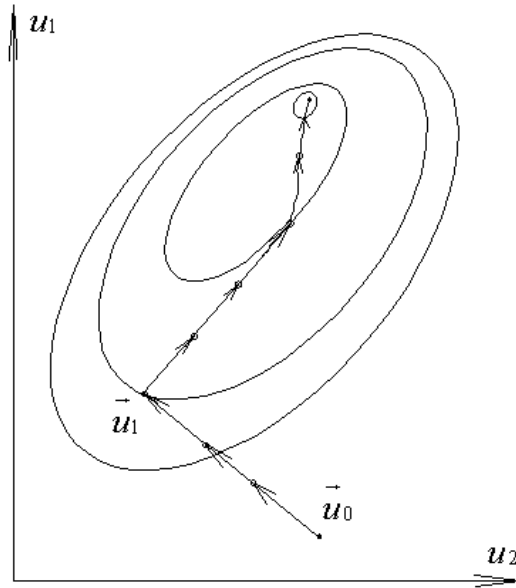


Рис. 4.6. Траєкторія руху за методом крутого спуску

4.1.4. Метод спряжених градієнтів (Флетчера-Рівса)

В розглянутих методах на кожному кроці ніяк не використовувалась інформація, отримана на попередніх кроках. Очевидно, використовуючи цю інформацію, можна розраховувати на прискорення його збіжності. Методи пошуку, в яких нове наближення формується на основі декількох попередніх, називаються багатокроковими. Одним з багатокрокових методів є двокроковий метод спряжених градієнтів. Ідея методу закладається в побудові наступного наближення за схемою:

$$\bar{u}_{k+1} = \bar{u}_k - h_k \nabla f(\bar{u}_k) + \beta_k (\bar{u}_k - \bar{u}_{k-1}), \quad (4.5)$$

де параметри h_k і β_k вибираються на кожному кроці оптимальним чином, тобто з умови

$$(h_k, \beta_k) = \arg \min_{h, \beta} f(\bar{u}_k - h \nabla f(\bar{u}_k) + \beta (\bar{u}_k - \bar{u}_{k-1})). \quad (4.6)$$

В загальному випадку знайти рішення даної задачі в явному вигляді не вдається, але для випадку квадратичної функції методу можна придати наступну форму:

$$\begin{aligned} \bar{u}_{k+1} &= \bar{u}_k + h_k \bar{s}_k, \quad \bar{s}_k = -\nabla f(\bar{u}_k) + \beta_k \bar{s}_{k-1}, \\ \beta_k &= \frac{|\nabla f(\bar{u}_k)|^2}{|\nabla f(\bar{u}_{k-1})|^2}, \quad \beta_0 = 0, \quad h_k = \arg \min_{h \geq 0} f(\bar{u}_k + h \bar{s}_k). \end{aligned} \quad (4.7)$$

Характерною особливістю методу є той факт, що він приводить до знаходження точки мінімуму квадратичної цільової функції за число ітерацій, що не перевищує розмірність задачі.

Не дивлячись на те, що останній алгоритм отримано для квадратичної цільової функції, метод може успішно застосовуватись і для неквадратичних функцій. Але тоді метод не буде кінцевим, тому при цьому видається доцільним через кожні n кроків проводити оновлення методу, тобто приймати $\beta = 0$ при $k = 0, n, 2n, \dots$. Метод Флетчера-Рівса є за формою методом першого порядку і тому простим в реалізації, вигідно відрізняється від звичайних градієнтних методів тим, що наділений по суті всіма перевагами методів другого порядку (в тому числі квадратичною збіжністю).

Приклад 4.2. Метод Флетчера-Рівса

Знайти точку мінімуму функції

$$f(x) = 4x_1^2 + 3x_2^2 - 4x_1x_2 + x_1,$$

якщо $\vec{x}^{(1)} = [0, 0]^T$.

Крок 1. Визначимо градієнт цільової функції $\nabla f(x) = [8x_1 - 4x_2 + 1, 6x_2 - 4x_1]^T$, тоді напрямок пошуку $\vec{s}^{(1)} = -\nabla f(\vec{x}^{(1)}) = -[1, 0]^T$.

Крок 2. Пошук вздовж прямої:

$$\vec{x}^{(2)} = \vec{x}^{(1)} - \alpha^{(1)} \nabla f(\vec{x}^{(1)}) \rightarrow \alpha_1 = 1/8,$$

$$\vec{x}^{(2)} = [0, 0]^T - 1/8 [1, 0]^T = [-1/8, 0]^T.$$

Крок 3. $k = 2$

$$\vec{s}^{(2)} = -[0, 1/2]^T - [1/4, 1][1, 0]^T = [-1/4, 1/2]^T.$$

Крок 4. Пошук вздовж прямої:

$$\vec{x}^{(3)} = \vec{x}^{(2)} + \alpha^{(2)} \vec{s}^{(2)} \rightarrow \alpha^{(2)} = 1/4,$$

$$\vec{x}^{(3)} = [-1/8, 0]^T + 1/4 [-1/4, 1/2]^T = [-3/16, -1/8]^T,$$

$$\nabla f(\vec{x}^{(3)}) = [0, 0]^T.$$

Таким чином, отримане рішення збігається з точним рішенням. Це пояснюється тим, що цільова функція квадратична, а похибки округлення відсутні.

4.2. Методи другого порядку

4.2.1. Метод Ньютона

Градієнтний метод заснований на припущенні про можливість локальної лінійної апроксимації цільової функції $f_0(\vec{u})$.

Якщо цільова функція двічі диференційована, то можна скористатися квадратичною апроксимацією цільової функції.

Розкладемо цільову функцію в ряд Тейлора:

$$f_0(\vec{u}_{k+1}) = f_0(\vec{u}_k) + \nabla f_0(\vec{u}_k)^T (\vec{u}_{k+1} - \vec{u}_k) + \frac{1}{2} (\vec{u}_{k+1} - \vec{u}_k)^T \cdot \nabla^2 f_0(\vec{u}_k) (\vec{u}_{k+1} - \vec{u}_k) \dots \quad (4.8)$$

Якщо знехтувати у цьому ряді всіма членами, починаючи із другого порядку, то необхідна умова екстремуму в точці \vec{u}_{k+1} буде мати вигляд:

$$\nabla f_0(\vec{u}_k) + \nabla^2 f_0(\vec{u}_k) (\vec{u}_{k+1} - \vec{u}_k) = 0 \quad (4.9)$$

Звідки:

$$\vec{u}_{k+1} = \vec{u}_k - [\nabla^2 f_0(\vec{u}_k)]^{-1} \nabla f_0(\vec{u}_k) \quad (4.10)$$

Останній вираз є алгоритмом методу Ньютона. У тому випадку, якщо \vec{u}_{k+1} буде лише приблизним розв'язком задачі оптимізації, необхідно прийняти \vec{u}_{k+1} в якості нової початкової точки і знову скористатися алгоритмом (4.10). Таким чином, метод Ньютона можна розглядати як послідовний пошук точок мінімуму квадратичних апроксимацій цільової функції $f_0(\vec{u})$.

Метод Ньютона має швидку збіжність поблизу екстремуму, де помилка, викликана нехтуванням членами вищого порядку в розкладанні $f_0(\vec{u})$ у ряд Тейлора, незначна. Для квадратичних цільових функцій метод Ньютона дозволяє отримати значення екстремуму за один крок.

Слід відмітити, що метод Ньютона може бути застосований для оптимізації функції однієї змінної $f(u)$, що двічі диференційована. Тоді алгоритм (4.10) набуває наступного вигляду:

$$u_{k+1} = u_k - \left(\frac{d^2 f}{du^2} \right)^{-1} \left(\frac{df}{du} \right). \quad (4.11)$$

Приклад 4.3. Метод Ньютона

Мінімізувати за допомогою методу Ньютона функцію

$$f(x) = 8x_1^2 + 4x_1x_2 + 5x_2^2,$$

$$\vec{x}^{(0)} = [10, 10]^T$$

Визначимо градієнт і матрицю Гессе для цієї функції:

$$\nabla f(x) = [16x_1 + 4x_2, 10x_2 + 4x_1]^T,$$

$$\nabla^2 f(x) = \begin{bmatrix} 16 & 4 \\ 4 & 10 \end{bmatrix}.$$

З формули (4.10) отримуємо:

$$\vec{x}^{(1)} = [10, 10]^T - \frac{1}{144} \begin{bmatrix} 10 & -4 \\ -4 & 16 \end{bmatrix} [200, 140]^T =$$

$$= [10, 10]^T - \frac{1}{144} [1440, 1440]^T,$$

тобто

$$\vec{x}^{(1)} = [0, 0]^T,$$

що збігається з точним рішенням.

4.2.2. Модифікований метод Ньютона

Досвід показує, що при дослідженні неквадратичних функцій метод Ньютона не відзначається високою надійністю. Насправді, коли початкова точка пошуку знаходиться на значній відстані від точки екстремуму, крок за методом Ньютона виявляється занадто великим, що може призвести до відсутності сходження. Метод можна доволі просто модифікувати таким чином, щоб забезпечити зменшення цільової функції від ітерації до ітерації та здійснювати пошук вздовж прямої, як в методі найшвидшого спуску. Алгоритм пошуку набуває наступного вигляду:

$$\vec{u}_{k+1} = \vec{u}_k - \alpha_k [\nabla^2 f_0(\vec{u}_k)]^{-1} \nabla f_0(\vec{u}_k). \quad (4.12)$$

Вибір α_k здійснюється таким чином, щоб

$$f_0(\vec{u}_{k+1}) \rightarrow \min;$$

це гарантує виконання нерівності

$$f_0(\vec{u}_{k+1}) \leq f_0(\vec{u}_k).$$

Такий метод носить назву модифікованого методу Ньютона і у випадках, коли обчислення точних значень перших і других похідних не викликає суттєвих труднощів, видається надійним і ефективним. Однак при використанні цього методу на кожному кроці виникає необхідність побудови і рішення лінійного рівняння, що містить елементи матриці Гессе $\nabla^2 f_0(\vec{u})$.

4.2.3. Метод Девідона-Флетчера-Пауела (ДФП).

Одним із недоліків методу Ньютона є необхідність обчислення зворотної матриці других похідних цільової функції по всіх незалежних змінних.

Існують методи, що дозволяють не обчислювати безпосередньо матрицю Гессе або зворотну до неї $[\nabla^2 f_0(\vec{u}_k)]^{-1}$, а апроксимувати її, використовуючи для цього знайдені градієнти в попередніх точках. У результаті такої апроксимації на деякій k -тій ітерації буде отримана

матриця H_k , що відповідає $[\nabla^2 f_0(\vec{u}_k)]^{-1}$. Завдяки цьому, метод ДФП можна віднести до методів другого порядку, хоча звичайно методи, що використовують для апроксимації зворотної матриці Гессе градієнти в попередніх точках, виділяють у самостійний клас так званих квазіньютонівських методів.

Метод ДФП є досить ефективним, тому що поєднує достоїнства методу найшвидшого спуску (швидка зміна цільової функції на перших ітераціях) і методу Ньютона (добра збіжність поблизу екстремуму).

Алгоритм методу ДФП має вигляд:

$$\vec{u}_{k+1} = \vec{u}_k - \gamma_k H_k \nabla f_0(\vec{u}_k), \quad (4.13)$$

$$H_{k+1} = H_k - \frac{H_k \vec{y}_k \vec{y}_k^T H_k}{\vec{y}_k^T H_k \vec{y}_k} + \frac{\vec{\delta}_k \vec{\delta}_k^T}{\vec{\delta}_k^T \vec{y}_k}, \quad H_1 = I, \quad (4.14)$$

де $\vec{y}_k = \nabla f_0(\vec{u}_{k+1}) - \nabla f_0(\vec{u}_k)$, $\vec{\delta}_k = \vec{u}_{k+1} - \vec{u}_k$, I – одинична матриця, γ_k – кроковий множник, який визначається з умови мінімізації функції в напрямку $H_k \nabla f_0(\vec{u}_k)$.

Видно, що при $k=1$ метод ДФП повністю збігається з методом найшвидшого спуску.

Надалі, за рахунок введення виправлень до H_k , метод ДФП буде наближатись до алгоритму (4.10).

Приклад 4.4. Метод ДФП

За допомогою метода ДФП знайти точку мінімуму функції

$$f(x) = 4x_1^2 + 3x_2^2 - 4x_1x_2 + x_1,$$

якщо $\vec{x}^{(1)} = [0, 0]^T$.

Крок 1. Нехай напрямок пошуку $\vec{s}^{(1)} = -\nabla f(\vec{x}^{(1)}) = -[1, 0]^T$.

Крок 2. Пошук вздовж прямої:

$$\vec{x}^{(2)} = \vec{x}^{(1)} - \alpha^{(1)} \nabla f(\vec{x}^{(1)}) \rightarrow \alpha_1 = 1/8,$$

$$\vec{x}^{(2)} = [0, 0]^T - 1/8 [1, 0]^T = [-1/8, 0]^T.$$

$$\text{Крок 3. } k=1, \quad H^{(2)} = H^{(1)} - \frac{H^{(1)} \vec{y}^{(1)} (\vec{y}^{(1)})^T H^{(1)}}{(\vec{y}^{(1)})^T H^{(1)} \vec{y}^{(1)}} + \frac{\vec{\delta}_1 \vec{\delta}_1^T}{\vec{\delta}_1^T \vec{y}_1},$$

$$H^{(1)} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = I,$$

$$\vec{\delta}^{(1)} = [-1/8, 0]^T - [0, 0]^T = [-1/8, 0]^T,$$

$$\begin{aligned}\bar{y}^{(k)} &= \left[0, \frac{1}{2}\right]^T - [1, 0]^T = \left[-1, \frac{1}{2}\right]^T, \\ -\frac{H^{(1)}\bar{y}^{(1)}\left(\bar{y}^{(1)}\right)^T H^{(1)}}{\left(\bar{y}^{(1)}\right)^T H^{(1)}\bar{y}^{(1)}} + \frac{\bar{\delta}_1 \bar{\delta}_1^T}{\bar{\delta}_1^T \bar{y}_1} &= \frac{\left[-\frac{1}{8}, 0\right]^T}{\left[-\frac{1}{8}, 0\right]} - \frac{\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \left[-1, \frac{1}{2}\right]^T \left[-1, \frac{1}{2}\right] \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}}{\left[-1, \frac{1}{2}\right] \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \left[-1, \frac{1}{2}\right]^T} = \\ &= \begin{bmatrix} \frac{1}{8} & 0 \\ 0 & 0 \end{bmatrix} - \begin{bmatrix} \frac{4}{5} & -\frac{2}{5} \\ -\frac{2}{5} & \frac{1}{5} \end{bmatrix}, \\ H^{(2)} &= \begin{bmatrix} 0,325 & \frac{2}{5} \\ \frac{2}{5} & \frac{4}{5} \end{bmatrix},\end{aligned}$$

Новий напрямок пошуку $\bar{s}^{(2)} = -H^{(2)}\nabla f\left(x^{(2)}\right) = -\left[\frac{1}{5}, \frac{2}{5}\right]^T$.

Крок 4. Пошук вздовж прямої:

$$\begin{aligned}\bar{x}^{(3)} &= \bar{x}^{(2)} - \alpha^{(2)}\bar{s}^{(2)} \rightarrow \alpha^{(2)} = \frac{5}{16}, \\ \bar{x}^{(2)} &= \left[-\frac{1}{8}, 0\right]^T - \frac{5}{16}\left[\frac{1}{5}, \frac{2}{5}\right]^T = \left[-\frac{3}{16}, -\frac{1}{8}\right]^T, \\ \nabla f\left(x^{(3)}\right) &= [0, 0]^T.\end{aligned}$$

Отже, пошук завершено, причому знайдена точка збігається з точним рішенням.

4.3. Застосування градієнтних методів в задачах адаптивного управління

Приклад 4.5. Задача адаптивної ідентифікації статичного об'єкта.

Змістовна постановка

Знайти рівняння, що описує залежність вихідної величини статичного об'єкта управління від вхідної, з використанням метода моделі, що самонастроюється (СНМ).

Математична постановка

Одномірний об'єкт описується рівнянням наступного вигляду:

$$y = a_1 u + a_2 u^2, \quad (4.15)$$

де y, u – вимірювані вихід і вхід об'єкта, a_1, a_2 – коефіцієнти, що треба визначити в результаті рішення задачі ідентифікації.

Модель об'єкта має таку ж структуру:

$$y_i = a_{1i} u + a_{2i} u^2 \quad (4.16)$$

де y_i – вихідна величина моделі, a_{1i}, a_{2i} – невідомі параметри моделі.

За критерій близькості виходу об'єкта і моделі обираємо квадрат миттєвої помилки між виходом об'єкта і моделі:

$$I = \varepsilon^2 = (y_i - y)^2 \quad (4.17)$$

Математичний опис задачі, що полягає в обчисленні коефіцієнтів a_{1i}, a_{2i} на кожному кроці ітераційної процедури оптимізації критерію оптимальності (4.17), має вигляд:

$$\begin{aligned} I = \varepsilon^2 &= (y_i - y)^2 \rightarrow \min_{a_{1i}, a_{2i}}, \\ y &= a_1 u + a_2 u^2, \\ y_i &= a_{1i} u + a_{2i} u^2. \end{aligned} \quad (4.18)$$

Початкові значення коефіцієнтів моделі $a_{1i}^{(0)}, a_{2i}^{(0)}$ визначаються за апріорними відомостями про об'єкт.

Структурна схема рішення задачі ідентифікації.

Згідно до методу СНП, структурна схема задачі ідентифікації має вигляд, представлений на рис. 4.7.

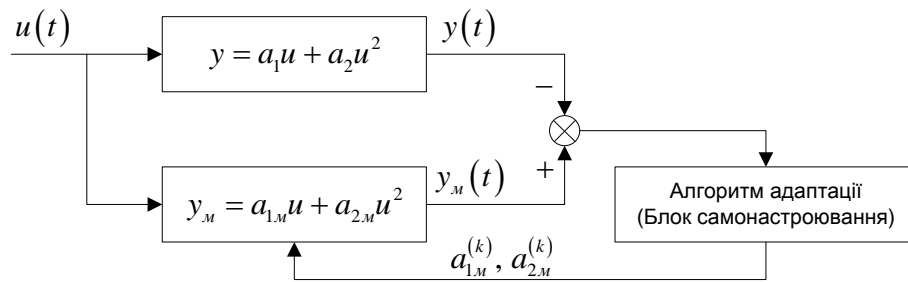


Рис. 4.7. Структурна схема системи ідентифікації статичного об'єкта

На вхід об'єкта і моделі подається вимірюваний сигнал $u(t)$. Помилка $\varepsilon(t)$ подається на блок обчислення a_{1i}, a_{2i} згідно до алгоритму адаптації, для синтезу якого використовується градієнтний метод оптимізації.

Синтез градієнтного алгоритму ідентифікації

Запишемо градієнтний алгоритм оптимізації для задачі (4.18).

$$a_{ii}^{(k+1)} = a_{ii}^{(k)} - h \frac{\partial I}{\partial a_{ii}}, \quad i = \overline{1, 2}. \quad (4.19)$$

Критерій оптимальності з урахуванням виразів, що описують об'єкт і модель, приймає вигляд

$$I = (a_{1i} + a_{2i} u^2 - a_1 u - a_2 u^2)^2. \quad (4.20)$$

Визначимо частинні похідні у виразах (4.19).

$$\frac{\partial I}{\partial a_1} = 2 \cdot \varepsilon \cdot u; \quad \frac{\partial I}{\partial a_2} = 2 \cdot \varepsilon \cdot u^2. \quad (4.21)$$

Підставимо (4.21) у вираз (4.19). Алгоритм адаптивної ідентифікації приймає вигляд

$$\begin{aligned} a_{1i}^{(k+1)} &= a_{1i}^{(k)} - 2h\epsilon u; \\ a_{2i}^{(k+1)} &= a_{2i}^{(k)} - 2h\epsilon u^2. \end{aligned} \quad (4.22)$$

Приклад 4.6. Задача синтезу дискретної адаптивної системи автоматичного управління з еталонною моделлю (ДАСАУЕМ)

Структурна схема ДАСАУЕМ зображена на рис. 4.8.

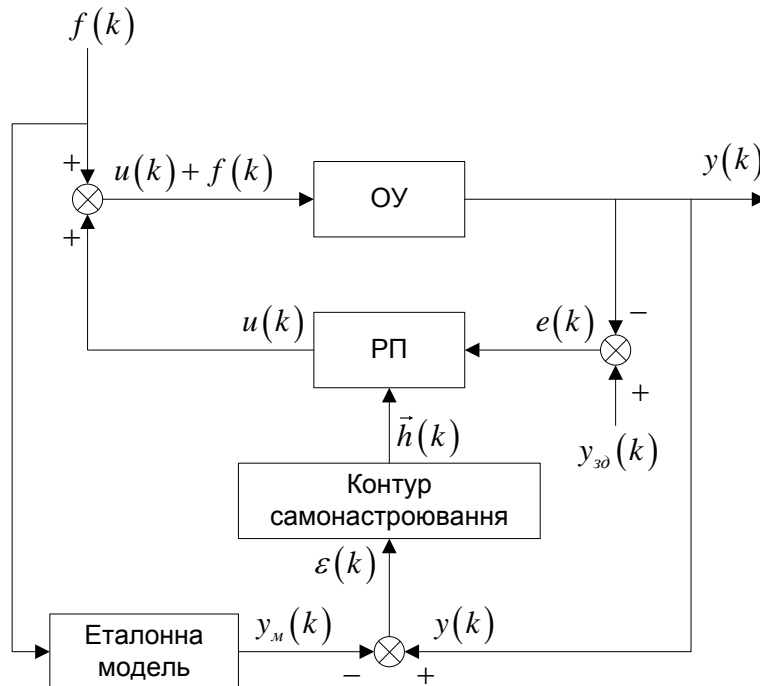


Рис. 4.8. Структурна схема ДАСАУЕМ

На рис. 4.8 прийнято наступні позначення:

ОУ – об’єкт управління зі змінними у часі параметрами;

РП – регулюючий пристрій (регулятор);

$y(k), y_i(k)$ – виходи об’єкта і моделі;

$e(k)$ – похибка регулювання;

$y_{\text{зад}}$ – завдання регулятору;

$\epsilon(k) = y(k) - y_i(k)$ – помилка між виходом системи автоматичного регулювання і еталонної моделі;

$\vec{h}(k)$ – вектор параметрів настроювання регулятора.

Регулюючий пристрій має настройки, що відповідають деякому номінальному режиму функціонування об’єкта управління. При зміні параметрів об’єкта настройки регулятора стають неоптимальними і вимоги до показників якості системи регулювання не виконуються. Контур

самонастроювання (КСН) на основі інформації про розбаланс $\varepsilon(k)$ коректує настройки регулятора РП таким чином, щоб $\varepsilon(k) \rightarrow 0$.

Змістовна постановка задачі синтезу ДАСАУЕМ

Знайти рівняння контуру самонастроювання, що забезпечує ефективну роботу системи регулювання при зміні параметрів об'єкта.

За допомогою еталонної моделі задається необхідний перехідний процес. при подачі збурення $f(k)$ виникають перехідні процеси в основній системі та в еталонній моделі. Якщо перехідні процеси відрізняються один від одного, то з'являється помилка $\varepsilon(k)$, що подається на КСН, вихідний сигнал якого змінює параметри регулятора до тих пір, поки $\varepsilon(k)$ не дорівнюватиме нулю, тоді $y(k) = y_i(k)$. Синтез КСН виконати з застосуванням градієнтного методу оптимізації критерію $I = \varepsilon^2(k+1)$.

Математична постановка задачі

Об'єкт управління описується рівнянням

$$y(k+1) + a_1 y(k) + a_2 y(k-1) = b u(k) + f(k). \quad (4.23)$$

Рівняння регулятора має вигляд

$$u(k) = -h_1 y(k) - h_2 y(k-1), \quad (4.24)$$

якщо прийняти, що $y_{\text{cxi}} = 0$. Тоді рівняння замкнутої АСР має наступний вигляд:

$$y(k+1) + (a_1 + b h_1) y(k) + (a_2 + b h_2) y(k-1) = f(k). \quad (4.25)$$

Рівняння еталонної моделі приймемо тої ж структури, що й рівняння основної АСР:

$$y_i(k+1) + a_{1i} y_i(k) + a_{2i} y_i(k-1) = f(k). \quad (4.26)$$

Таким чином, задача синтезу ДАСАУЕМ приймає вигляд

$$\begin{aligned} I &= \varepsilon^2(k+1) \rightarrow \min_{h_1, h_2}; \\ y(k+1) + (a_1 + b h_1) y(k) + (a_2 + b h_2) y(k-1) &= f(k); \\ y_i(k+1) + a_{1i} y_i(k) + a_{2i} y_i(k-1) &= f(k); \\ \varepsilon(k) &= y(k) - y_i(k) = \\ &= (a_1 + b h_1) y(k) + (a_2 + b h_2) y(k-1) - a_{1i} y_i(k) - a_{2i} y_i(k-1). \end{aligned} \quad (4.27)$$

Рішення задачі синтезу ДАСАУЕМ

Застосуємо градієнтний алгоритм оптимізації для рішення задачі (4.27). Тоді

$$\begin{aligned} h_1^{(k+1)} &= h_1^{(k)} - \lambda \frac{\partial I}{\partial h_1}; \\ h_2^{(k+1)} &= h_2^{(k)} - \lambda \frac{\partial I}{\partial h_2}; \end{aligned}$$

Враховуючи вирази для I та $\varepsilon(k+1)$ в (4.27), отримаємо

$$\frac{\partial I}{\partial h_1} = \frac{\partial \varepsilon^2(k+1)}{\partial h_1} = 2 \cdot \varepsilon(k+1) \cdot \frac{\partial \varepsilon(k+1)}{\partial h_1};$$

$$\frac{\partial I}{\partial h_2} = \frac{\partial \varepsilon^2(k+1)}{\partial h_2} = 2 \cdot \varepsilon(k+1) \cdot \frac{\partial \varepsilon(k+1)}{\partial h_2}.$$

В даному випадку

$$\frac{\partial \varepsilon(k+1)}{\partial h_1} = by(k); \quad \frac{\partial \varepsilon(k+1)}{\partial h_2} = by(k-1).$$

Звідси остаточно отримаємо алгоритм контуру самонастроювання

$$\begin{aligned} h_1^{(k+1)} &= h_1^{(k)} - 2\lambda \varepsilon(k+1) by(k) \\ h_2^{(k+1)} &= h_2^{(k)} - 2\lambda \varepsilon(k+1) by(k-1) \end{aligned} \tag{4.28}$$

4.4. Контрольні завдання

1. В задачі безумовної оптимізації виконати один крок пошуку екстремуму простим градієнтним методом і обчислити значення цільової функції в знайденій точці.

$$I(u) = u^2 + 2u + 2 \rightarrow \min_u.$$

Пошук починати з точки $u^{(0)} = 1$. Параметр кроку в алгоритмі пошуку взяти $h=0,25$.

2. В задачі безумовної оптимізації виконати один крок градієнтного спуску і обчислити значення цільової функції $I(u_1^{(1)}, u_2^{(1)})$.

$$I(u) = u_1^2 + u_2^2 \rightarrow \min_{u_1, u_2};$$

$$u_i^{(1)} = u_i^{(0)} - h \frac{\partial I(u_1^{(0)}, u_2^{(0)})}{\partial u_i};$$

Початкові умови: $u_1^{(0)} = 4$; $u_2^{(0)} = 4$.

Параметр кроку: $h = 1$

3. Задача безумовної оптимізації функції має вигляд:

$$I(u) = u^2 + 2u + 2 \rightarrow \min_u.$$

Обчислити оптимальний крок методом Ньютона в алгоритмі градієнтного спуску.

4. Задано функцію

$$f(x) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2$$

і дві перших точки, отриманих в процесі пошуку мінімуму функції $f(x)$:

$$x^{(0)} = [-1, 2; 1]^T, \quad x^{(1)} = [-1, 3; 1, 07]^T.$$

Визначте напрямок пошуку з точки $x^{(1)}$, користуючись наступними методами:

- а) методом найшвидшого спуску,
- б) методом Ньютона.

5. Побудувати лінію рівня функції $f(u_1, u_2) = (u_1 - 2)^2 + 4(u_2 - 1)^2 = 10$. Обчислити та побудувати градієнт в одній із точок лінії рівня.

6. Знайти мінімум функції $f(x_1, x_2) = x_1^2 - x_1 + x_2^2 - x_2 + 1$.

7. Знайти точки стаціонарності функції $f(x_1, x_2) = x_1 - x_1^2 - x_1^3 - x_1 x_2$ і вияснити, які з них є точками максимуму.

8. Записати матрицю Гессе функції задачі 7 в точках її стаціонарності.

9. При апроксимації експериментально отриманих значень $y_i^A(x_i)$ лінійною залежністю за методом найменших квадратів розв'язується наступна екстремальна задача вибору коефіцієнтів a_0 і a_1 :

$$f_0(a_0, a_1) = \sum_i (a_0 + a_1 x_i - y_i^A(x_i))^2 \rightarrow \min_{a_0, a_1};$$

а) записати умови оптимальності для цієї задачі;

б) записати вираз для матриці Гессе цільової функції і показати, що матриця позитивно визначена.

в) знайти коефіцієнти a_0 і a_1 , якщо результати експерименту задані таблицею:

x_i	1	2	3	4
y_i^A	2	3	5	8

10. Виконати один цикл пошуку максимуму функції

$$f_0(x_1, x_2) = x_1 + x_2 - (x_1 x_2)^2 - x_1^2 - 2x_2^2$$

з початкової точки $\vec{x}^{(0)} = (1, 0)$, використовуючи наступні методи:

а) метод Коші;

б) градієнтний метод зі сталим кроком $h = 1$.

5. МЕТОДИ НУЛЬОВОГО ПОРЯДКУ БАГАТОМІРНОЇ ОПТИМІЗАЦІЇ

Методи нульового порядку застосовуються у тих випадках, коли з якихось причин визначення градієнта цільової функції неможливе. Вони використовуються у тих випадках, коли функція f_0 задана алгоритмічно, зокрема, коли для обчислення значень функції за тих чи інших значеннях аргументу потрібно провести натурний або числовий експеримент. Ці методи придатні, коли функція f_0 недиференційована або її значення визначаються з похибкою, що призводить до великих неточностей при обчисленні похідних.

Методи прямого пошуку є методами, у яких використовуються лише значення функції.

Розглянемо функцію двох змінних. Її лінії постійного рівня наведені на рис. 5.1, а мінімум знаходиться в точці (u_1^*, u_2^*) . Найпростішим методом пошуку є *метод по координатного спуску*. Із точки **A** ми робимо пошук мінімуму вздовж напрямку осі u_1 і, таким чином, знаходимо точку **B**, в якій дотична до лінії постійного рівня паралельна осі u_1 . Потім, роблячи пошук із точки **B** у напрямку осі u_2 , отримуємо точку **C**, роблячи пошук паралельно осі u_1 , отримуємо точку **D**, і т.д. Таким чином, ми приходимо до оптимальної точки. Будь-який із одномірних методів, описаних у попередній главі, може бути використаний тут для пошуку вздовж осі. Таким чином цю ідею можна застосувати для функцій n змінних.

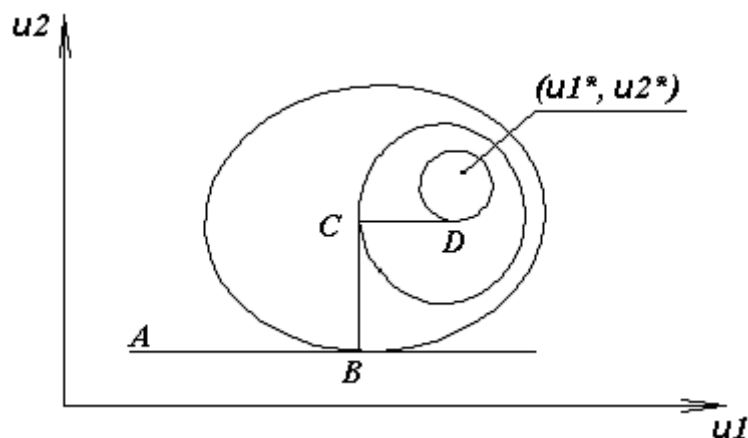


Рис. 5.1. Траєкторія руху методом по координатного спуску

Теоретично даний метод ефективний у випадку єдиного мінімуму функції. Але на практиці він виявляється занадто повільним. Тому були розроблені більш складні методи, що використовують більше інформації на підставі вже отриманих значень функції.

Було запропоновано кілька функцій, які через свої властивості є тестовими для таких методів. Нижче наведено кілька прикладів таких функцій.

Функція Розенброка:

$$f(u_1, u_2) = 100(u_2 - u_1^2)^2 + (1 - u_1)^2; u^* = (1; 1). \quad (5.1)$$

Функція Пауела:

$$f(u) = (u_1 + 10u_2)^2 + 5(u_3 - u_4)^2 + (u_2 - 2u_3)^4 + 10(u_1 - u_4)^2; u^* = (0; 0; 0; 0). \quad (5.2)$$

Двовимірна експонентна функція:

$$f(u_1, u_2) = \sum_a \left[(e^{-au_1} - e^{-au_2}) - (e^{-a} - e^{-10a}) \right], \text{ де } a = 0, 1, (0, 1) \text{ і } 1^*; u^* = (1; 10). \quad (5.3)$$

Будь-яка серйозна оптимізаційна процедура повинна ефективно вирішувати задачі (5.1), (5.2), (5.3) та інші тестові задачі.

5.1. Метод Гауса-Зейделя.

Розглянута вище стратегія пошуку покладена в основу методу Гауса-Зейделя. Так, для задачі $f(\vec{u}) \rightarrow \min_{\vec{u}}$, де $\vec{u} \in R^n$ алгоритм пошуку можна записати таким чином:

$$u_i^{onm} = \arg \min_{u_i} f(u_1^{onm}, u_2^{onm}, \dots, u_i, \dots, u_{n-1}, u_n)$$

Основна ідея цього методу полягає в тому, що пошук екстремуму здійснюється за допомогою почергової зміни незалежних змінних вздовж однієї з координатних осей.

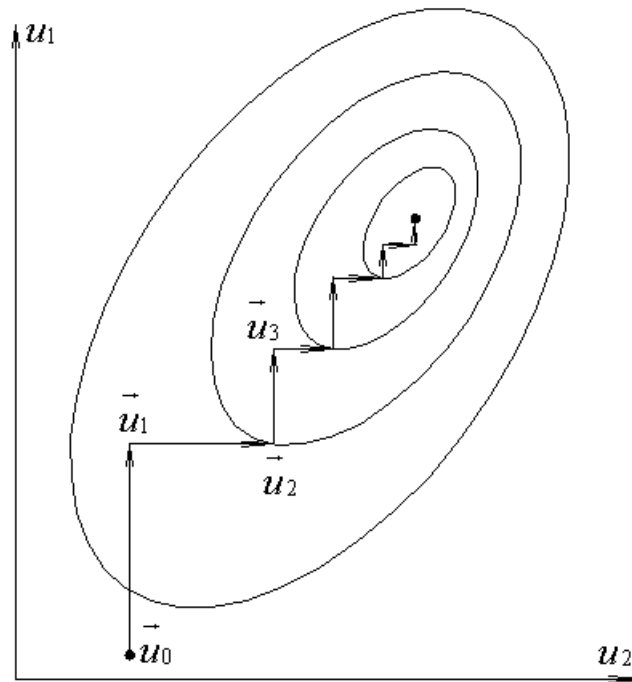


Рис. 5.2. Траєкторія руху методом Гауса-Зейделя

Порядок дій згідно алгоритму можна записати так:

1) Вибір вихідної точки \vec{u}_k , $k=0$ і першого напрямку пошуку \vec{e}_j , $j=1, \dots, m$, наприклад, $\vec{e}_1 = \{1, 0, 0, \dots, 0\}^T$ – одиничний вектор, що збігається у напрямку з віссю u_1 .

2) Визначення $\vec{u}_{k+1} = \vec{u}_k + h_k \vec{e}_j$, де h_k – кроковий множник, обраний з умови мінімізації $f_0(\vec{u})$ по напрямку \vec{e}_j .

3) Якщо $j < n$, де n – число незалежних змінних, то присвоюється $k=j$, $j=j+1$ і виконується пункт 2.

Якщо $j=n$, то присвоюється $\vec{u}_k = \vec{u}_0$, $k=0$ і виконується пункт 1.

Траєкторія руху цим методом для функції двох змінних зображена на рис. 4.6.

Процес пошуку припиняється, якщо на жодному з напрямків уздовж координатних осей не спостерігається зменшення цільової функції.

Ефективність методу Гауса-Зейделя істотно залежить від спрямованості ліній рівня цільової функції. Так, для цільової функції, зображеної на рис. 5.3, застосування методу Гауса-Зейделя неефективно, а для цільової функції, зображеної на рис. 5.4, метод Гауса-Зейделя забезпечує знаходження екстремуму всього за два кроки.

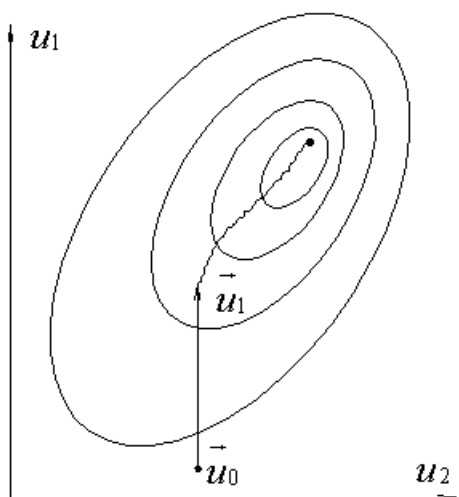


Рис. 5.3. Випадок неефективного використання метода Гауса-Зейделя

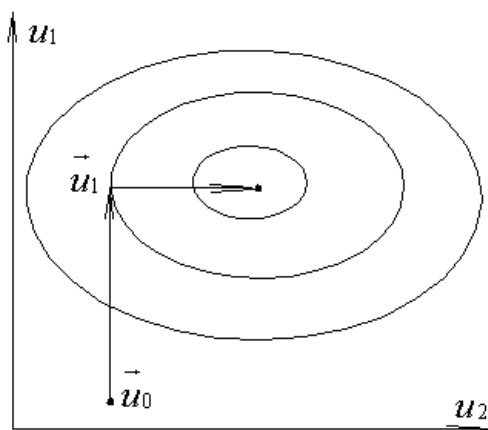


Рис. 5.4. Випадок ефективного використання метода Гауса-Зейделя

5.2. Метод Розенброка.

Метод Розенброка заснований на тому, що замість безперервного пошуку по координатах, що відповідають напрямкам незалежних змінних, після кожного циклу координатного пошуку здійснюється поворот системи координат таким чином, що одна з координат збігається з напрямком усього кроку в попередньому циклі координатного пошуку.

Метод Розенброка по суті своїй зводиться до знаходження такої системи координат, у якій напрямки пошуку витягуються вздовж головних осей квадратичної апроксимації цільової функції.

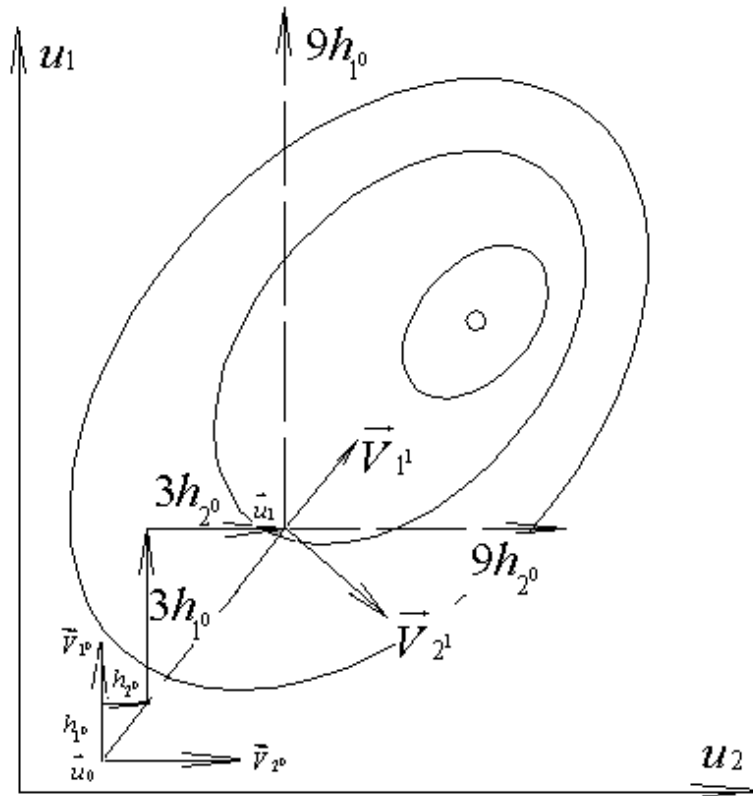


Рис. 5.5. Траєкторія руху методом Розенброка

Алгоритм методу для пошуку максимуму має вигляд (рис. 5.5):

1. Вибір вихідної точки \vec{u}_k , $k = 0$.
2. Вибір ортогональних напрямків пошуку \vec{V}_{jk} , $j = 1, \dots, m$ і величин кроків h_{jk} по кожному з напрямків пошуку. Як ортогональні напрямки пошуку \vec{V}_{j0} вибираються напрямки, що відповідають напрямкам координатних осей $\vec{V}_{j0} = \vec{e}_j$, $j = \overline{1, m}$.
3. Введення допоміжної величини S_j , $j = 1, \dots, m$. Присвоєння $S_j = 0$.
4. Здійснення пошукових кроків по всіх ортогональних напрямках:

$$\vec{u} = \vec{u}_k + h_{jk} \vec{V}_{jk}.$$

- 4.1. Якщо $j \geq m$, то присвоєння $j = 1$ і перехід до наступного пункту, якщо $j < m$, то $j = j + 1$ і виконання пункту 4.1.

- 4.2. Присвоєння $\vec{u}_k = \vec{u}$, $S_j = S_j + h_{jk}$, $j = 1, \dots, m$ і вибір нових величин кроку:

$$h_{jk} = \begin{cases} \alpha \cdot h_{jk}, & \text{якщо } f_0(\vec{u}) > f_0(\vec{u}_k), \quad j = 1, \dots, m, \\ \beta \cdot h_{jk}, & \text{якщо } f_0(\vec{u}) < f_0(\vec{u}_k), \quad j = 1, \dots, m. \end{cases} \quad (5.4)$$

Рекомендується приймати значення $\alpha = 3$; $\beta = -\frac{1}{2}$.

Перехід до пункту 4.1.

Ці операції необхідно виконувати доти, поки цільова функція не перестане поліпшуватися. У результаті буде досягнута точка \vec{u}_{k+1} .

5. Визначення нових напрямків пошуку (нових ортогональних векторів) $\vec{V}_{j,k+1}$.

5.1. Введення додаткових векторів \vec{W}_j :

$$\vec{W}_j = \sum_{j=1}^n S_j \vec{V}_{jk}, j = 1, \dots, m. \quad (5.5)$$

5.2. Обчислення одиничних векторів:

$$\begin{aligned} \vec{V}_{1,k+1} &= \frac{\vec{W}_1}{\|\vec{W}_1\|} \\ \vec{V}_{j,k+1} &= \frac{\vec{W}_j - \sum_{i=1}^{j-1} (\vec{W}_j^T \vec{V}_{i,k+1}) \vec{V}_{i,k+1}}{\left\| \vec{W}_j - \sum_{i=1}^{j-1} (\vec{W}_j^T \vec{V}_{i,k+1}) \vec{V}_{i,k+1} \right\|}, j = 2, 3, \dots, m. \end{aligned} \quad (5.6)$$

5.3. Присвоєння:

$$\begin{aligned} \vec{V}_{j,k} &= \vec{V}_{j,k+1}, j = 1, \dots, m \\ S_j &= 0, j = 1, \dots, m \end{aligned} \quad (5.7)$$

і перехід до пункту 4.

Процес пошуку переривається, коли величина \vec{W}_1 після численних послідовних наближень залишається малою, а напрямок пошуку залишається незмінним. Очевидно, що \vec{W}_1 є єдиним вектором від \vec{u}_k до \vec{u}_{k+1} .

На рис. 5.6 показана траєкторія пошуку при мінімізації функції Розенброка методом Розенброка (числа позначають кількість обчислень цільової функції).

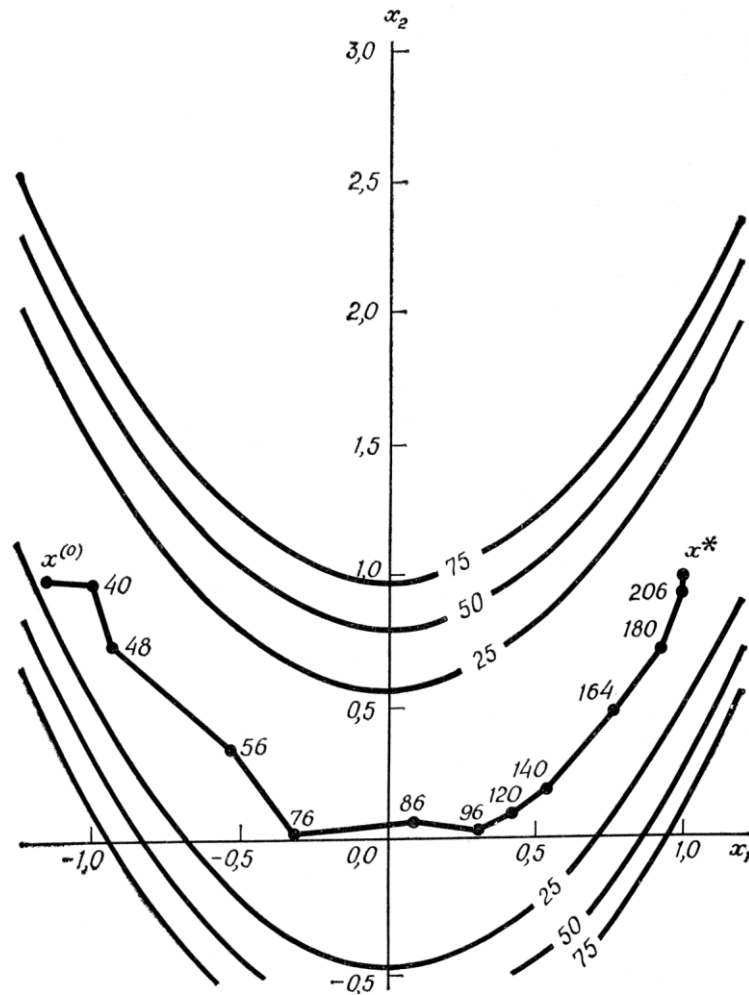


Рис. 5.6. Траєкторія пошуку при мінімізації функції Розенброка методом Розенброка

Приклад 5.1. Ілюстрація роботи алгоритму Розенброка

Мінімізувати $f(x) = 4(x_1 - 5)^2 + (x_2 - 6)^2$. Початкова точка $x^{(0)} = \begin{bmatrix} 8 \\ 9 \end{bmatrix}$.

Розв'язок. У точці $x^{(0)}$ значення функції $f(x^{(0)}) = 45,000$; $h_1 = h_2 = 0,10$.
Напрями початкового пошуку співпадають з координатними вісями x_1 та x_2 : $\vec{V}_1^{(0)} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$, $\vec{V}_2^{(0)} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$.

Спочатку обрахуємо $f(x)$ у точці

$$x = \begin{bmatrix} 8,00 + 0,10 \cdot 1 \\ 9,00 + 0,10 \cdot 0 \end{bmatrix} = \begin{bmatrix} 8,10 \\ 9,00 \end{bmatrix}.$$

Тут $f(x) = 47,44$, тобто має місце невдача. Потім обрахуємо $f(x)$ у точці

$$x = \begin{bmatrix} 8,00 + 0,10 \cdot 0 \\ 9,00 + 0,10 \cdot 1 \end{bmatrix} = \begin{bmatrix} 8,00 \\ 9,10 \end{bmatrix}.$$

Тут $f(x) = 45,61$ и знову має місце невдача. Таким чином, на наступному циклі і h_1 , і h_2 мають бути помножені на $\beta = -\frac{1}{2}$, або $h_3 = h_4 = -0,50 \cdot 0,10 = -0,05$. Крок здійснюється з останнього вдалого значення x , тобто з точки $x = \begin{bmatrix} 8 \\ 9 \end{bmatrix}$.

Спочатку обрахуємо значення $f(x)$ у точці

$$x = \begin{bmatrix} 8,00 - 0,05 \cdot 1 \\ 9,00 - 0,05 \cdot 0 \end{bmatrix} = \begin{bmatrix} 7,95 \\ 9,00 \end{bmatrix}.$$

Тут $f(x) = 43,81$, що свідчить про успіх. Потім обрахуємо $f(x)$ у точці

$$x = \begin{bmatrix} 7,95 - 0,05 \cdot 0 \\ 9,00 - 0,05 \cdot 1 \end{bmatrix} = \begin{bmatrix} 7,95 \\ 8,95 \end{bmatrix},$$

де $f(x) = 43,125$, що знову свідчить про успіх. На наступному циклі h помножується на 3, тобто $h_5 = h_6 = 3 \cdot (-0,05) = -0,15$.

Нижче наведені декілька послідовних циклів першого етапу.

Номер пошуку	h	x_1	x_2	$f(x)$	Успіх (S) або невдача (F)
5	-0,15	7,80	8,95	40,06	S
6	-0,15	7,80	8,80	39,20	S
7	-0,45	7,35	8,80	29,93	S
8	-0,45	7,35	8,35	27,61	S
9	-1,35	6,00	8,35	9,522	S
10	-1,35	6,00	7,00	5,000	S
12	-4,05	1,95	7,00	32,21	F
13	-4,05	6,00	2,95	13,30	F

Тепер вже у кожному координатному напрямі за успіхом відбулася невдача, і, таким чином, закінчився нульовий етап пошуку. Потім обраховуються нові напрями пошуку так, щоб $\vec{V}_1^{(1)}$ був напрямлений вздовж вектора, що йде з точки $x^{(0)} = \begin{bmatrix} 8 \\ 9 \end{bmatrix}$ у $x^{(1)} = \begin{bmatrix} 6 \\ 7 \end{bmatrix}$, при чому остання точка відповідає найкращому значенню $f(x)$, одержаному на нульовому етапі; $\vec{V}_2^{(1)}$ ортонормований до $\vec{V}_1^{(1)}$. Вектори $\vec{W}_1^{(0)}$ та $\vec{W}_2^{(0)}$ вираховуються за формулою (5.5), а $\vec{V}_1^{(1)}$ та $\vec{V}_2^{(1)}$ - за формулою (5.6).

$$\vec{W}_1^{(0)} = \begin{bmatrix} -2 \\ -2 \end{bmatrix}, \vec{W}_2^{(0)} = \begin{bmatrix} 0 \\ -2 \end{bmatrix}, \vec{V}_1^{(1)} = \frac{\begin{bmatrix} -2 \\ -2 \end{bmatrix}}{\sqrt{(-2)^2 + (-2)^2}} = \begin{bmatrix} -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix}^T,$$

$$\vec{V}_2^{(1)} = \frac{\begin{bmatrix} 0 & -2 \end{bmatrix}^T - \begin{bmatrix} 0 & -2 \end{bmatrix} \begin{bmatrix} -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix}^T \begin{bmatrix} -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix}^T}{\left\| \begin{bmatrix} 0 & -2 \end{bmatrix}^T - \begin{bmatrix} 0 & -2 \end{bmatrix} \begin{bmatrix} -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix}^T \begin{bmatrix} -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix}^T \right\|} = \frac{\begin{bmatrix} 1 & -1 \end{bmatrix}^T}{\sqrt{1^2 + (-1)^2}} = \begin{bmatrix} \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} \end{bmatrix}.$$

У першому циклі на першому етапі пошуку ($k=1$) приймемо $h_1 = h_2 = -4,05 \cdot (-0,5) = 2,025$. Спочатку обрахуємо $f(x)$ у напрямку пошуку $\vec{V}_1^{(1)}$ у точці

$$x = \begin{bmatrix} 6,000 + (2,025) \cdot (-0,706) \\ 6,000 + (2,025) \cdot (-0,706) \end{bmatrix} = \begin{bmatrix} 4,568 \\ 5,568 \end{bmatrix}.$$

Тут $f(x) = 0,9327$, що свідчить про успіх. Потім обрахуємо $f(x)$ у напрямку пошуку $\vec{V}_2^{(1)}$ у точці

$$x = \begin{bmatrix} 4,568 + (2,025) \cdot 0,706 \\ 5,568 + (2,025) \cdot (-0,706) \end{bmatrix} = \begin{bmatrix} 6,000 \\ 4,136 \end{bmatrix}.$$

Тут $f(x) = 7,474$, що свідчить про невдачу. Решта даних для першого етапу наведена нижче:

Номер пошуку	h	x_1	x_2	$f(x)$	Успіх (S) або невдача (F)
3	6,075	0,272	1,272	11,175	F
4	— 1,125	3,852	6,284	5,351	F
5	— 3,038	6,716	7,716	14,722	F
6	0,506	4,926	5,210	0,646	S
7	1,518	3,852	4,136	8,743	F
8	1,518	6,000	4,136	7,473	F

Оскільки рух у напрямку 1 призводить до успіху в перших двох випадках, а у напрямку 2 – до невдачі, то у пошуках 3 та 4 беремо $h_3 = 3 \cdot 2,025 = 6,075$ і $h_4 = -0,5 \cdot 2,025 = -1,0125$ відповідно. Після невдачі у пошуках 3 та 4 $h_3 = 6,075$ та $h_4 = -1,0125$ помножуються на $-0,5$, що дає відповідно h_5 та h_6 . Після пошуку 6 мав місце успіх по обом напрямкам, а після пошуку 8 відбулися дві послідовні невдачі, тому у відповідності до даного алгоритму на цьому кроці етап 1 завершується. На цьому етапі

завершимо розгляд даного прикладу. Проте наведена вище процедура була продовжена, поки критерій збіжності не був задоволений (що потребувало 111 обчислень цільової функції, а відносне відхилення її значення від правильного значення, рівного 0, становило $5,5 \cdot 10^{-11}$, відносне відхилення x_1 становило $3,0 \cdot 10^{-6}$, а відхилення x_2 становило $4,4 \cdot 10^{-6}$).

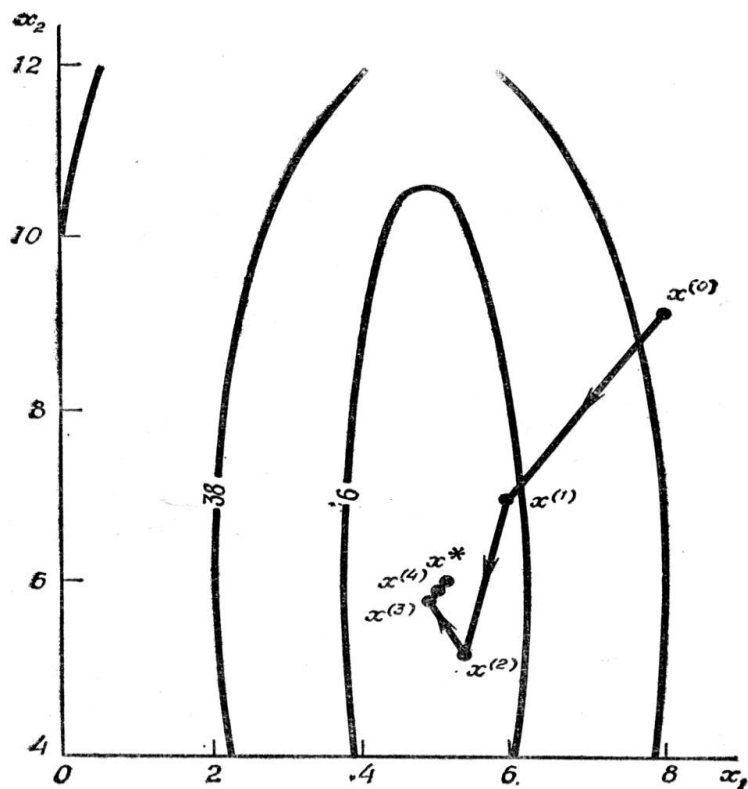


Рис. 5.7. Траєкторія основної частини мінімізації на перших чотирьох етапах

На рис. 5.7 показана траєкторія основної частини мінімізації на перших чотирьох етапах (34 обчислення функції); після проведення цих етапів були отримані $x^{(4)} = [5,036 \ 5,938]^T$ та $f(x^{(4)}) = 9,46 \cdot 10^{-3}$. Інші 77 обчислень функції знадобились для збільшення точності x та $f(x)$.

5.3. Метод пошуку по симплексу.

Перші спроби розв'язку оптимізаційних задач без обмежень на основі прямого пошуку пов'язані з використанням одномірних методів оптимізації. Як правило, при реалізації таких методів допустима область визначення показника якості функціонування системи (цільової функції) замінюється дискретною множиною (гратами) точок простору керованих змінних, а потім використовуються різні стратегії зменшення області, що

містить розв'язок задачі. Часто ця процедура виявляється еквівалентною рівномірному пошуку у вузлах ґрат і, отже, непридатною для розв'язку задач із числом змінних, що перевищує 2. Більш корисна ідея полягає у виборі базової точки й оцінюванні значень цільової функції в точках, що оточують базову. Наприклад, при розв'язку задачі із двома змінними можна скористатися квадратним зразком, зображеним на рис. 4.12. Потім «найкраща» з п'яти досліджуваних точок вибирається в якості наступної базової точки, навколо якої будується аналогічний зразок. Якщо жодна з кутових точок не має переваги перед базовою, то розміри зразка варто зменшити, після чого продовжити пошук.

Одна із стратегій пошуку, що викликає особливий інтерес покладена в основу *методу пошуку по симплексу*, запропонованого Спендлі, Хекстом і Хімсвортом. Процедура симплексного пошуку Спендлі, Хекста і Хімсворта базується на тому, що експериментальним зразком, що містить найменшу кількість точок, є регулярний симплекс. Регулярний симплекс в N -мірному просторі є багатогранником, утвореним $N+1$ рівновіддаленими одна від одної точками-вершинами. Наприклад, у випадку двох змінних симплексом є рівносторонній трикутник; у тривимірному просторі симплекс – тетраедр. В алгоритмі симплексного пошуку використовується важлива властивість симплексів, відповідно до якої новий симплекс можна побудувати на будь-якій грані початкового симплекса шляхом переносу обраної вершини на належну відстань уздовж прямої, проведеної через центр ваги інших вершин початкового симплекса. Отримана в такий спосіб точка є вершиною нового симплекса, а обрана при побудові вершина початкового симплексу відкидається. Неважко побачити, що при переході до нового симплексу потрібно одне обчислення значення цільової функції. Рис. 4.13 ілюструє процес побудови нового симплексу на площині.

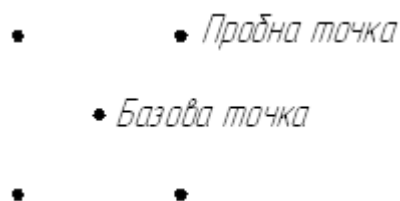


Рис. 5.8. Квадратний зразок (частковий випадок кубічного зразку)

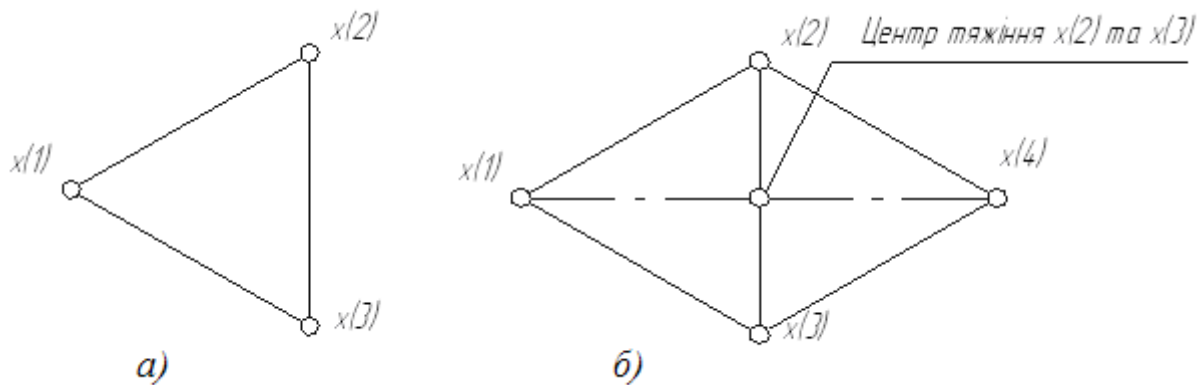


Рис. 5.9. Побудова нового симплексу: а) - початковий симплекс; б) - новий симплекс

Робота алгоритму симплексного пошуку починається з побудови регулярного симплекса у просторі незалежних змінних й оцінювання значень цільової функції у кожній з вершин симплексу. При цьому визначається вершина, якій відповідає найбільше значення цільової функції. Потім знайдена вершина проектується через центр ваги інших вершин симплексу в нову точку, що використовується як вершина нового симплексу. Якщо функція спадає досить плавно, ітерації тривають доти, поки або не буде накрита точка мінімуму, або не почнеться циклічний рух по двох або більше симплексах. У таких ситуаціях можна скористатися наступними трьома правилами.

Правило 1. «Накриття» точки мінімуму

Якщо вершина, якій відповідає найбільше значення цільової функції, побудована на попередній ітерації, то замість неї береться вершина, якій відповідає наступне по величині значення цільової функції.

Правило 2. Циклічний рух

Якщо деяка вершина симплексу не виключається протягом більш ніж M ітерацій, то необхідно зменшити розміри симплексу за допомогою коефіцієнта редукції і побудувати новий симплекс, обравши в якості базової точку, якій відповідає мінімальне значення цільової функції. Спендлі, Хекст і Хімсворт запропонували обчислювати M по формулі:

$$M = 1.65N + 0.05N^2,$$

де N - розмірність задачі, а M округляється до найближчого цілого числа. Для застосування даного правила потрібно встановити величину коефіцієнту редукції.

Правило 3. Критерій закінчення пошуку

Пошук завершується, коли або розміри симплекса, або різниці між значеннями функції у вершинах стають досить малими. Щоб можна було застосовувати ці правила, необхідно задати величину параметру закінчення пошуку.

Реалізація досліджуваного алгоритму заснована на обчисленнях двох типів: (1) побудові регулярного симплексу при заданих базовій точці й масштабному множнику і (2) розрахунку координат відбитої точки. Побудова симплексу є досить простою процедурою, тому що з елементарної геометрії відомо, що при заданих початковій (базовій) точці $x(0)$ і масштабному множнику α координати інших N вершин симплекса в N -мірному просторі обчислюються за формулою:

$$x(i) = \begin{cases} x_j(0) + \delta_1, & \text{якщо } j \neq i, \\ x_j(0) + \delta_2, & \text{якщо } j = i, \end{cases} \quad (5.8)$$

для i та $j=1,2,3,\dots,N$.

Прирости δ_1 і δ_2 , що залежать лише від N і обраного масштабного множника α , визначаються по формулах:

$$\delta_1 = \left[\frac{(N+1)^{1/2} + N - 1}{N\sqrt{2}} \right] \alpha, \quad (5.9)$$

$$\delta_2 = \left[\frac{(N+1)^{1/2} + N - 1}{N\sqrt{2}} \right] \alpha, \quad (5.10)$$

Помітимо, що величина масштабного множника α вибирається дослідником, виходячи з характеристик розв'язуваної задачі. При $\alpha=1$ ребра регулярного симплексу мають одиничну довжину.

Обчислення другого типу, пов'язані з відбиттям щодо центра ваги, також представляють собою нескладну процедуру. Нехай $x(j)$ - точка, що підлягає відбиттю. Центр ваги інших N точок розташований у точці:

$$x_c = \frac{1}{N} \sum_{\substack{i=0 \\ i \neq j}}^N x(i). \quad (5.11)$$

Всі точки прямої, що проходить через $x(j)$ і x_c , задаються формулою:

$$x = x(j) + \lambda(x_c - x(j)) \quad (5.12)$$

При $\lambda=0$ отримуємо вихідну точку $x(j)$, тоді як значення $\lambda=1$ відповідає центру ваги x_c . Для того щоб побудований симплекс мав властивість регулярності, відбиття повинне бути симетричним. Отже, нова вершина знаходиться при $\lambda=2$. Таким чином:

$$x(j)_{\text{нов}} = 2x_c - x(j)_{\text{попереднє}} \quad (5.13)$$

Проілюструємо обчислювальну схему методу наступним прикладом.

Приклад 5.2. Обчислення відповідно до методу пошуку по симплексу

Мінімізувати $f(x) = (1-x_1)^2 + (2-x_2)^2$.

Розв'язок. Для побудови вихідного симплексу потрібно задати початкову точку і масштабний множник. Нехай $x^{(0)} = [0,0]^T$ і $\alpha=2$. Тоді:

$$\delta_1 = \left[\frac{\sqrt{3}+1}{2\sqrt{2}} \right] \alpha = 1.9318$$

$$\delta_2 = \left[\frac{\sqrt{3}-1}{2\sqrt{2}} \right] \alpha = 0.5176$$

Використовуючи ці два параметри, обчислимо координати двох інших вершин симплексу:

$$x^{(1)} = [0+0,5176; 0+1,9318]^T = [0,5176; 1,9318]^T,$$

$$x^{(2)} = [0+1,9318; 0+0,5176]^T = [1,9318; 0,5176]^T,$$

яким відповідають значення цільової функції $f(x^{(1)}) = 0,2374$ і $f(x^{(2)}) = 3,0658$. Тому що $f(x^{(0)})=5$, необхідно відбити точку $x^{(0)}$ щодо центра ваги двох інших вершин симплексу:

$$x_c = \frac{1}{2} \sum_{i=1}^2 x^{(i)} = \frac{1}{2} (x^{(1)} + x^{(2)}).$$

Використовуючи формулу (4.22), отримуємо:

$$x^{(3)} = x^{(1)} + x^{(2)} - x^{(0)},$$

$$x^T = [2,4494; 2,4494]^T.$$

В отриманій точці $f(x^{(3)}) = 2,3027$, тобто спостерігається зменшення цільової функції. Новий симплекс утворений точками $x^{(1)}$, $x^{(2)}$ і $x^{(3)}$. Відповідно до алгоритму варто відбити точку $x^{(2)}$, якій відповідає найбільше значення цільової функції, щодо центра ваги точок $x^{(1)}$ і $x^{(3)}$. Ітерації тривають доти, поки не знадобиться застосування правил 1, 2 і 3, які були сформульовані вище.

Розглянутий вище алгоритм має кілька очевидних переваг:

1. Розрахунки й логічна структура методу відрізняються порівняльною простотою, і, отже, відповідна програма для ЕОМ виявляється короткою.
2. Рівень вимог до об'єму пам'яті ЕОМ невисокий, масив має розмірність $(N+1, N+2)$.
3. Використовується порівняно невелике число заздалегідь установлених параметрів: масштабний множник α , коефіцієнт зменшення множника α (якщо застосовується правило 2) і параметри закінчення пошуку.
4. Алгоритм виявляється ефективним навіть у тих випадках, коли помилка обчислення значень цільової функції велика, оскільки при його реалізації оперують найбільшими значеннями функції у вершинах, а не найменшими.

Перераховані фактори характеризують метод пошуку по симплексу як досить корисний при проведенні обчислень у реальному часі.

Алгоритм має також ряд істотних недоліків.

1. Не виключене виникнення труднощів, пов'язаних із масштабуванням, оскільки всі координати вершин симплексу залежать від того самого масштабного множника α . Щоб обійти такі труднощі, у практичних задачах треба промасштабувати всі змінні для того, щоб їх значення були приблизно однаковими по величині.
2. Алгоритм працює занадто повільно, тому що отримана на попередніх ітераціях інформація не використовується для прискорення пошуку.
3. Не існує простого способу розширення симплексу, що не вимагає перерахунку значень цільової функції у всіх точках зразка. Таким чином, якщо з якої-небудь причини цільова функція зменшується (наприклад, якщо зустрічається область із вузьким «яром» або «хребтом»), то пошук повинен тривати із зменшеною величиною кроку.

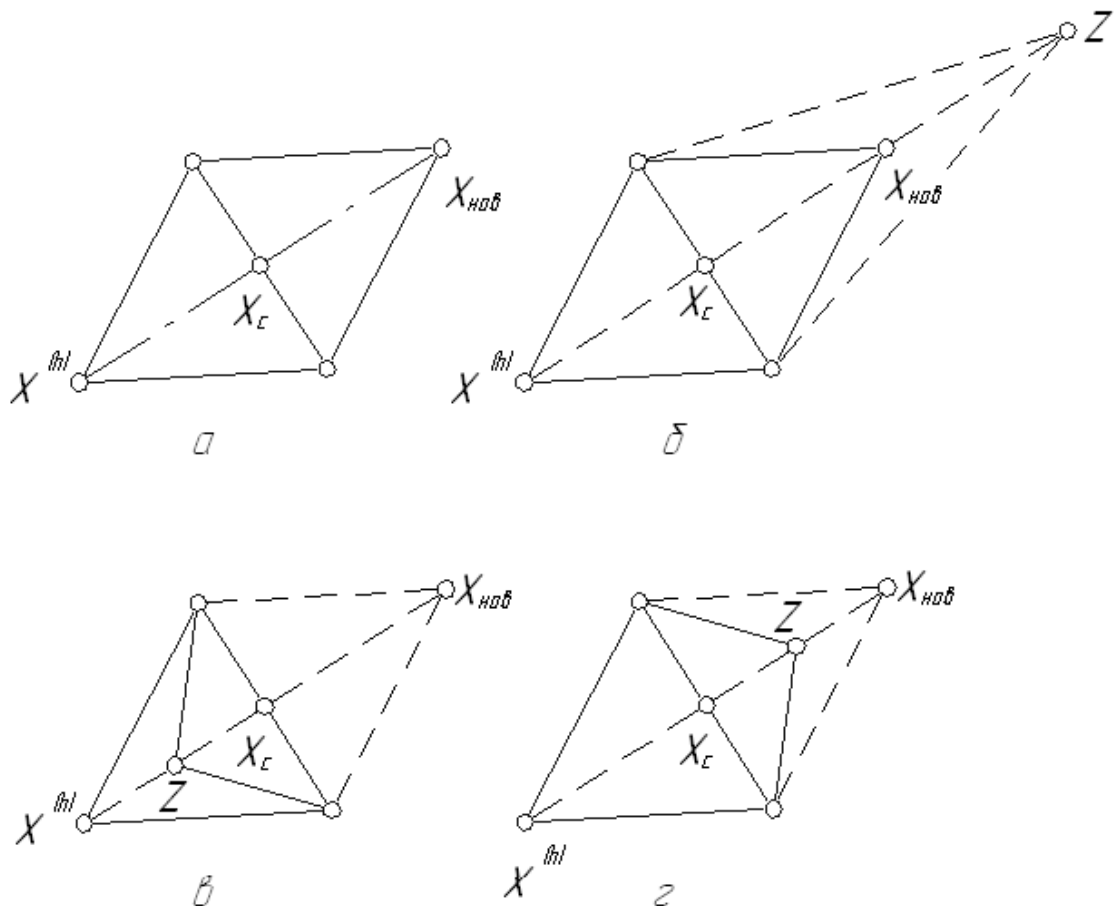


Рис. 5.10. Розтяг і стиск симплексу

а – нормальне відбиття, б – розтяг, в – стиск, г – стиск.

Модифікована процедура пошуку по симплексу, розроблена Нелдером і Мідом, частково усуває деякі з перерахованих недоліків. Неважко помітити, що хоча формула для визначення вершин регулярного симплекса виявляється досить зручною при побудові вихідного зразка, однак вагомим підстав для збереження властивості регулярності симплексу в процесі пошуку немає. Отже, при відбитті симплекса існує можливість як його розтягування, так і стискання. При розрахунках по методу Нелдера і Міда використовуються вершини симплексу $x^{(h)}$ (якій відповідає найбільше значення цільової функції $f^{(h)}$), $g^{(h)}$ (якій відповідає наступне по величині значення цільової функції $f^{(g)}$) і $x^{(i)}$ (якій відповідає найменше значення цільової функції $f^{(i)}$). Нагадаємо, що відбиття вершини симплексу здійснюється по прямій:

$$x = x^{(h)} + \lambda(x_c - x^{(h)}), \quad (5.14)$$

$$x = x^{(h)} + (1 + \theta)(x_c - x^{(h)}). \quad (5.15)$$

При $\theta = 1$ має місце так назване нормальне відбиття симплексу, оскільки точка $x_{нов.}$ розташовується на відстані $\|x_c - x^{(j)}\|$ від точки x_c . Якщо $-1 \leq \theta < 1$, то спостерігається стисле відбиття, або *стиск* симплекса, тоді як вибір $\theta > 1$ забезпечує розтягнуте відбиття, або *розтягування* симплексу. На рис. 5.8 наведені можливі варіанти відбиття. Три значення параметра θ , використовувані при нормальному відбитті, стиску і розтягуванні, позначаються через α , β , і γ відповідно. Реалізація методу починається з побудови вихідного симплексу і визначення точок $x^{(h)}$, $x^{(g)}$, $x^{(i)}$ і x_c . Після нормального відбиття здійснюється перевірка значень цільової функції за критерієм закінчення пошуку в точках відбитого симплексу. Якщо пошук не закінчений, то за допомогою тестів, наведених на рис. 5.8, вибирається одна з операцій – нормальне відбиття, розтягування або стиск. Ітерації тривають, поки зміни значень цільової функції у вершинах симплексу не стануть незначними. Як задовільні значення параметрів α , β , і γ Нелдер і Мід рекомендують використовувати $\alpha = 1$, $\beta = 0,5$, і $\gamma = 2$.

Результати окремих чисельних експериментів показують, що метод Нелдера - Міда має достатню ефективність і високу надійність в умовах наявності випадкових збурень або помилок при визначенні значень цільової функції. В 1969 р. Бокс і Дрейпер стверджували, що цей метод є «найбільш ефективним із всіх відомих методів послідовної оптимізації». В 1972 р. Паркінсон і Хатчінсон досліджували вплив вибору параметрів α , β , γ і способу побудови вихідного симплексу на ефективність пошуку. Вони встановили, що орієнтація вихідного симплексу на відміну від його форми є істотним чинником, що впливає на процедуру пошуку, і запропонували використати значення параметрів $(\alpha; \beta; \gamma) = (2; 0,25; 2,5)$.

Такий вибір параметрів дозволив забезпечити гарну роботу алгоритму при повторенні послідовних розтягувань симплексу.

5.4. Метод Хука-Дживса.

Пошук згідно цього методу складається із послідовності *кроків пошуку* для дослідження функції навколо базисної точки, з якої у випадку успіху робиться *пошук за зразком*.

Опис цієї процедури наведено нижче:

- А. Вибрати початкову базисну точку b_1 і крок довжиною h_j для кожної змінної x_j , $j=1,2,\dots,n$. Довжина кроку може бути обрана однаковою для всіх змінних.
- В. Досліджувати $f(X)$ навколо базисної точки b_1 з метою отримання даних про локальне поведіння функції $f(X)$. Ці дані будуть використовуватися для знаходження потрібного напрямку пошуку за зразком, за допомогою якого можна сподіватися досягти більшого спадання функції. *Пошук, що досліджує, навколо базисної точки b_1 виконується таким чином:*
 - 1) Обчислити значення функції $f(b_1)$ у базисній точці b_1 .
 - 2) Кожна змінна по черзі змінюється додаванням довжини кроку. Таким чином, ми обчислюємо значення функції $f(b_1+h_1e_1)$, де e_1 – одиничний вектор у напрямку осі x_1 . Якщо це призводить до зменшення значення функції, то b_1 замінюється на $b_1+h_1e_1$. У протилежному випадку обчислюємо значення функції $f(b_1-h_1e_1)$, і якщо її значення зменшується, то b_1 замінюємо на $(b_1-h_1e_1)$. Якщо жоден із виконаних кроків не призводить до зменшення значення функції, то точка b_1 залишається незмінною і розглядаються зміни в напрямку осі x_2 , тобто знаходиться значення функції $f(b_1+h_2e_2)$ і т.д. Коли будуть розглянуті всі n змінних, ми будемо мати нову базисну точку b_2 .
 - 3) Якщо $b_2=b_1$, тобто зменшення функції не було досягнуто, то дослідження повторюється навколо тієї ж базисної точки b_1 , але із зменшеною довжиною кроку. На практиці задовільним є зменшення кроку (кроків) у десять разів від початкової довжини.
 - 4) Якщо $b_2 \neq b_1$, то виконується *пошук за зразком*.
- С. При пошуку за зразком використовується інформація, отримана в процесі дослідження, і мінімізація функції здійснюється пошуком у напрямку, заданому зразком. Ця процедура виконується в такий спосіб:
 - 1) Розумно рухатися із базисної точки b_2 у напрямку (b_2-b_1) , оскільки пошук у цьому напрямку вже привів до зменшення

значення функції. Тому обчислимо функцію в точці зразку: $P_1 = b_1 + 2(b_2 - b_1)$.

У загальному випадку: $P_i = b_i + 2(b_{i+1} - b_i)$.

- 2) Потім дослідження варто продовжувати навколо точки P_1 (P_i).
- 3) Якщо найменше значення на кроці С.2) менше значення в базисній точці b_2 (у загальному випадку b_{i+1}), то отримують нову базисну точку b_3 (у загальному випадку b_{i+2}), після чого варто повторити крок С.1). У протилежному випадку не робити пошук за зразком із точки b_2 (b_{i+1}), а продовжити дослідження в точці b_2 (b_{i+1}).
- 4) Завершити цей процес, коли довжина кроку (довжини кроків) буде зменшена до заданого малого значення.

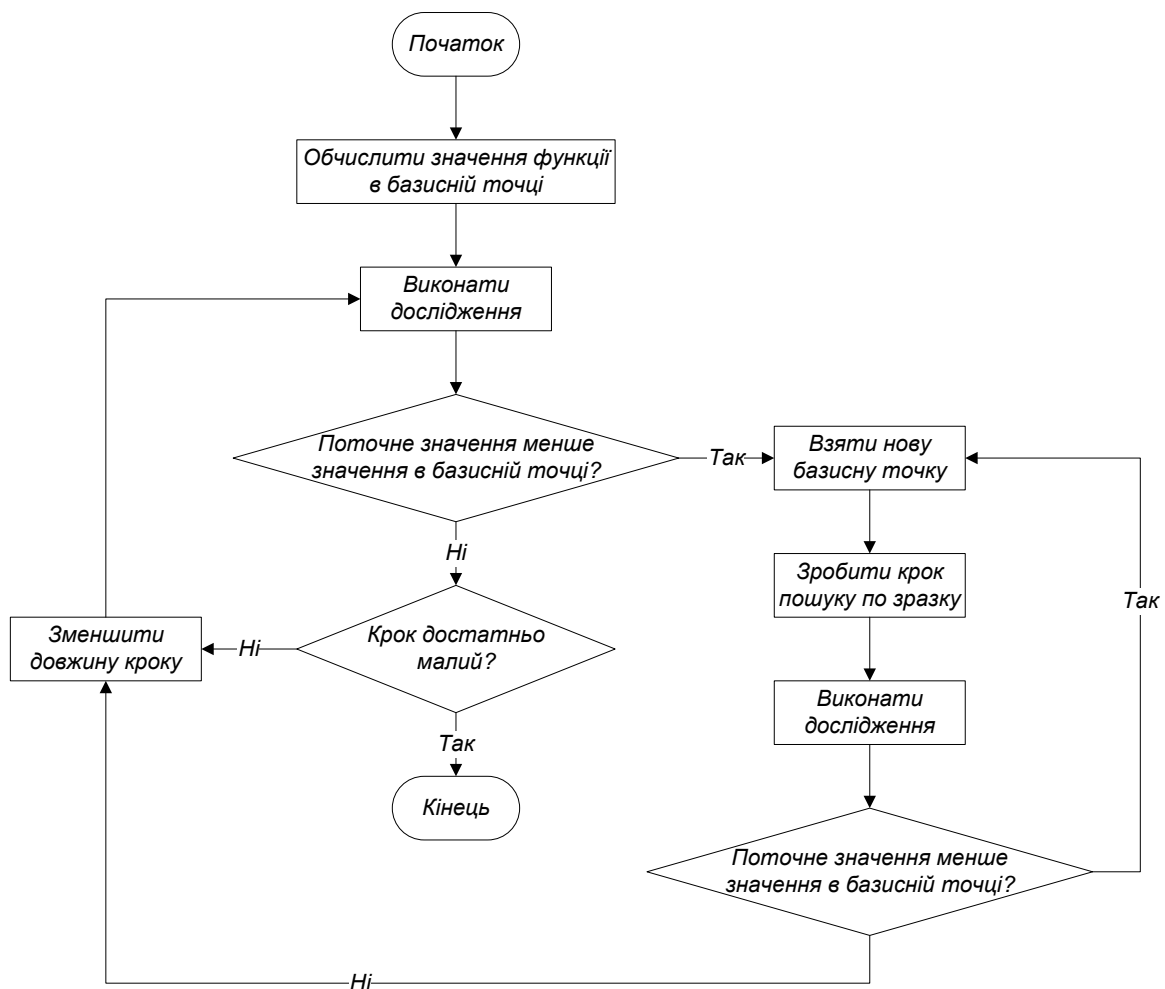


Рис. 5.11. Інформаційна схема метода Хука-Дживса

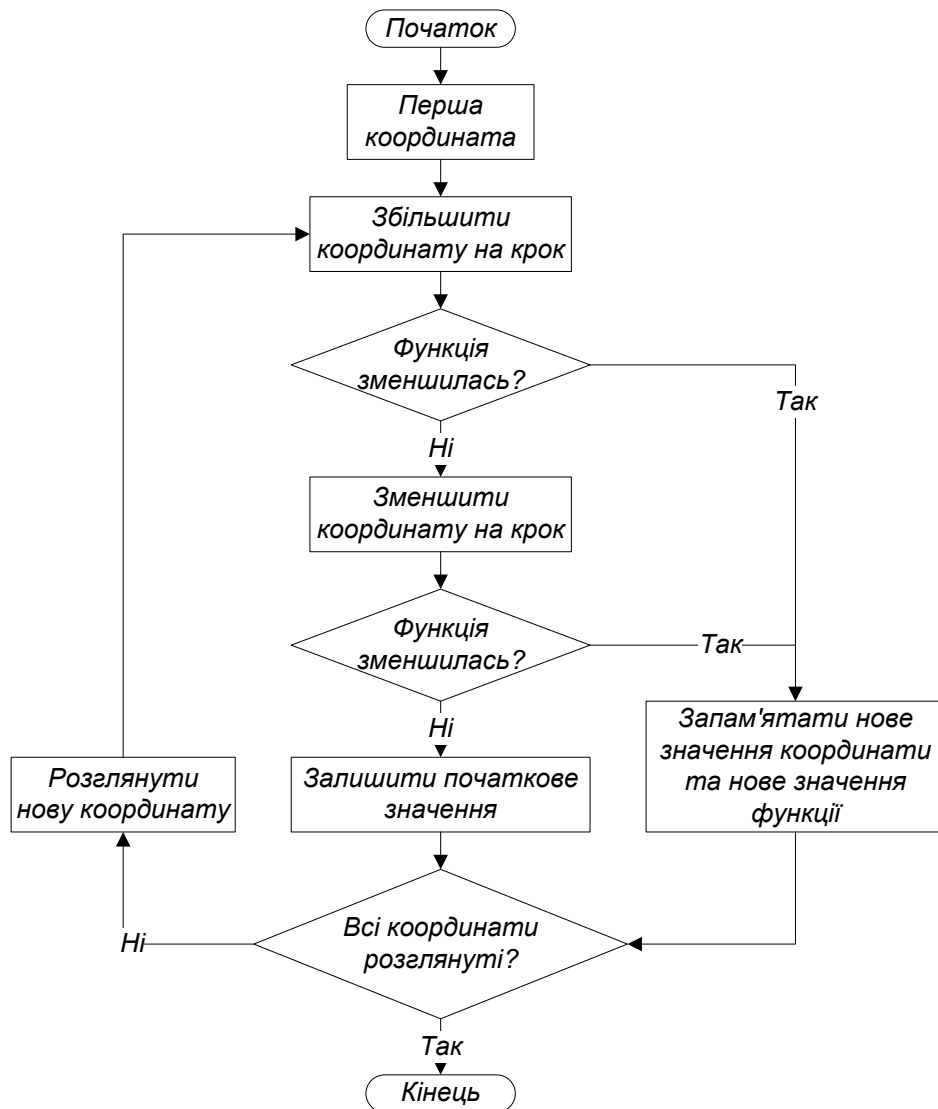


Рис. 5.12. Інформаційна схема досліджуючого пошуку

Приклад 5.3. Розв'язок задачі оптимального параметричного синтезу регулятора методом Хука-Дживса.

Однією із важливих задач оптимізації в теорії управління є задача оптимального параметричного синтезу регулятора. Цю задачу можна розв'язати методами моделювання і оптимізації для регуляторів різноманітної структури, а також для різноманітних об'єктів керування – лінійних, із запізненням, нелінійних та ін.

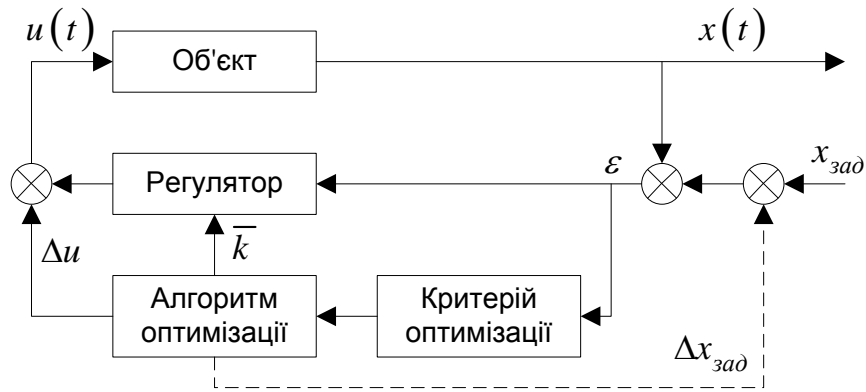


Рис. 5.13. Структурна схема метода моделювання та оптимізації одноконтурної САР

Структурна схема методу моделювання та оптимізації одноконтурної САР (системи автоматичного регулювання) зображена на рис. 5.13.

Моделюючи процес, знаходимо сигнал розбалансу $e(t)$ і використовуємо його для знаходження інтегрального критерію оптимізації. Математична постановка задачі оптимального параметричного синтезу АСР по інтегральному критерію в загальному вигляді може бути записана так:

$$I = \int_{t_H}^{t_K} f_0(\bar{x}, \bar{u}, \bar{g}, \bar{e}) dt \rightarrow \min_{k \in V_k}$$

$$\dot{\bar{x}} = \bar{f}(\bar{x}, \bar{u}, \bar{b}) + W(t)$$

$$\bar{y} = \bar{\varphi}(x, \bar{c})$$

$$\bar{u} = \bar{\psi}(\bar{x}, \bar{k})$$

$$\bar{u} \in V_u, \bar{x} \in V_x, k \in V_k$$

\bar{b}, \bar{c} – вектори параметрів об'єкту і вимірювального приладу;

\bar{k} – вектор параметрів регулятора

V_k, V_u, V_x – відкриті і закриті частини;

$W(t)$ – детерміноване збурення.

Для рішення цієї задачі використовують методи нульового порядку. Розглянемо задачу ОПС пропорційно-інтегрального регулятора для об'єкта першого порядку із запізненням.

Змістовна постановка задачі

Необхідно визначити параметри динамічної настройки пропорційно-інтегрального регулятора для відпрацювання заданого сигналу при наявності обмежень на керування.

Математична постановка задачі

Передаточна функція об'єкту:

$$W_{об} = \frac{Y(p)}{U(p)} = \frac{2 \cdot e^{-30 \cdot p}}{100 \cdot p + 1}$$

Інтегральний критерій має вигляд: $I = \int_{t_n}^{t_k} e^2 \cdot t^\alpha \cdot dt \rightarrow \min_{K_p, T_u}$

Передаточна функція регулятора:

$$W_p = K_p \left(1 + \frac{1}{T_u \cdot p} \right), \quad \bar{k} = \begin{bmatrix} K_p \\ T_u \end{bmatrix}$$

Обмеження на керування:

$$u_{\min} \leq u \leq u_{\max}.$$

Процедуру оптимізації виконати з використанням алгоритму Хука-Дживса.

Для знаходження значень K_p і T_u методом Хука-Дживса подається збурення $\Delta x_{зад}$. Сигнал розбалансу поступає на блок, що обчислює критерій оптимізації та передає одержану величину на блок, що реалізує алгоритм оптимізації. Останній коректує параметри настройки регулятора, \bar{k} . Потім подається нове збурення і дії повторюються. Оптимізація продовжується доти, доки зміна значення критерію оптимальності не стає меншою за задану точність, тобто $|I^{(k)} - I^{(k-1)}| < \varepsilon$, що свідчить про досягнення екстремуму.

Процес пошуку оптимальних настройок регулятора методом Хука-Дживса можна реалізувати програмно. На рис. 5.14 наведено розрахунок настройок регулятора з використанням програми „Параметрический синтез одноконтурной АСР.v.1.1”, розробленої Бунке А.С. (інтерфейс програми на російській мові). Початкові значення параметрів настройки регулятора одержані інженерною методикою: $K_p = 1.17$, $T_u = 70 \text{ с}$. Кінцеві значення після восьми кроків оптимізації становлять: $K_p = 0.87$, $T_u = 105 \text{ с}$.

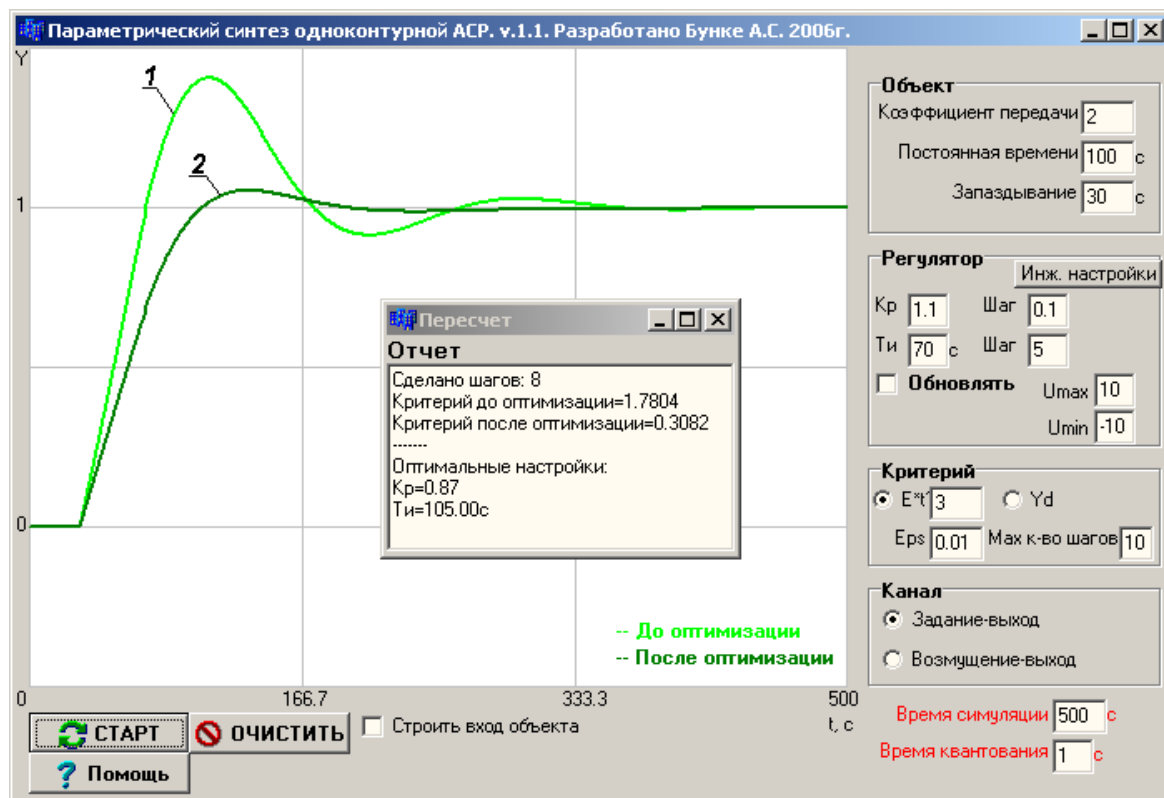


Рис. 5.14. Розрахунок оптимальних параметрів регулятора методом моделювання і оптимізації

1 – перехідний процес до оптимізації; 2 – перехідний процес після восьми кроків оптимізації

Приклад 5.4. Розв’язок задачі оптимального параметричного синтезу регуляторів каскадної САУ.

Розглянемо задачу синтезу стабілізуючого та коректуючого ПІ-регуляторів каскадної САУ методом Хука-Дживса. Система має вигляд, зображений на рис. 5.15.

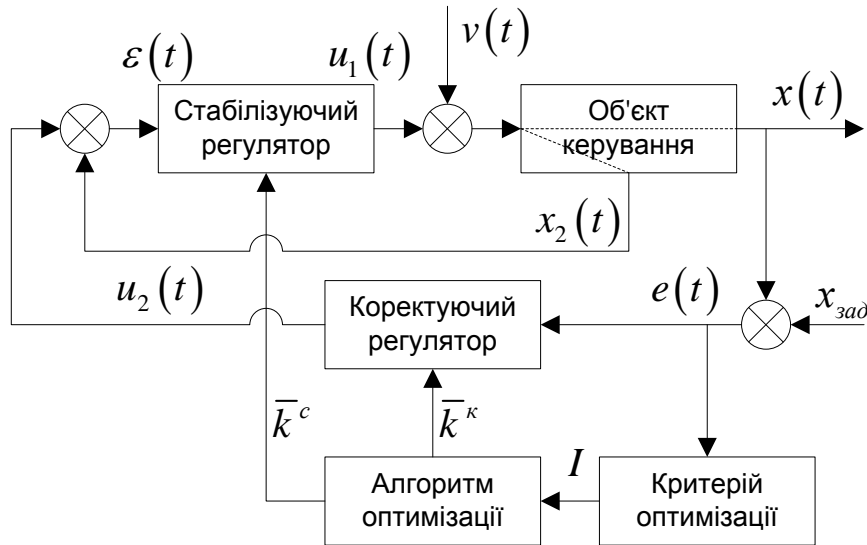


Рис. 5.15. Структурна схема методу моделювання та оптимізації каскадної САР

Математична постановка задачі

Передаточна функція об'єкту: $W_{об}^{інерц.} = \frac{1 \cdot e^{-15 \cdot p}}{150 \cdot p + 1}$, $W_{об}^{опер.} = \frac{1 \cdot e^{-0.5 \cdot p}}{5 \cdot p + 1}$

Інтегральний критерій має вигляд: $I = \int_{t_u}^{t_k} e^2 \cdot t^\alpha \cdot dt \rightarrow \min_{K_p, T_u}$

Передаточна функція регулятора: $W_p^\kappa = K_p^\kappa \left(1 + \frac{1}{T_u^\kappa \cdot p} \right)$, $\bar{k}^\kappa = \begin{bmatrix} K_p^\kappa \\ T_u^\kappa \end{bmatrix}$,

$$W_p^c = K_p^c \left(1 + \frac{1}{T_u^c \cdot p} \right), \quad \bar{k}^c = \begin{bmatrix} K_p^c \\ T_u^c \end{bmatrix}$$

На рис. 5.16 наведено розрахунок настройок регулятора з використанням програми „Параметрический синтез ПИ для Каскадной системы”, розробленої Бунке А.С. (інтерфейс програми на російській мові). Початкові значення параметрів настройки регуляторів одержані інженерною методикою: $K_p^k = 7$, $T_u^k = 105\text{ c}$; $K_p^c = 7$, $T_u^c = 3.5\text{ c}$. Кінцеві значення після 30 кроків пошуку становлять: $K_p^k = 5.2$, $T_u^k = 150\text{ c}$; $K_p^c = 8.5$, $T_u^c = 0.5\text{ c}$.

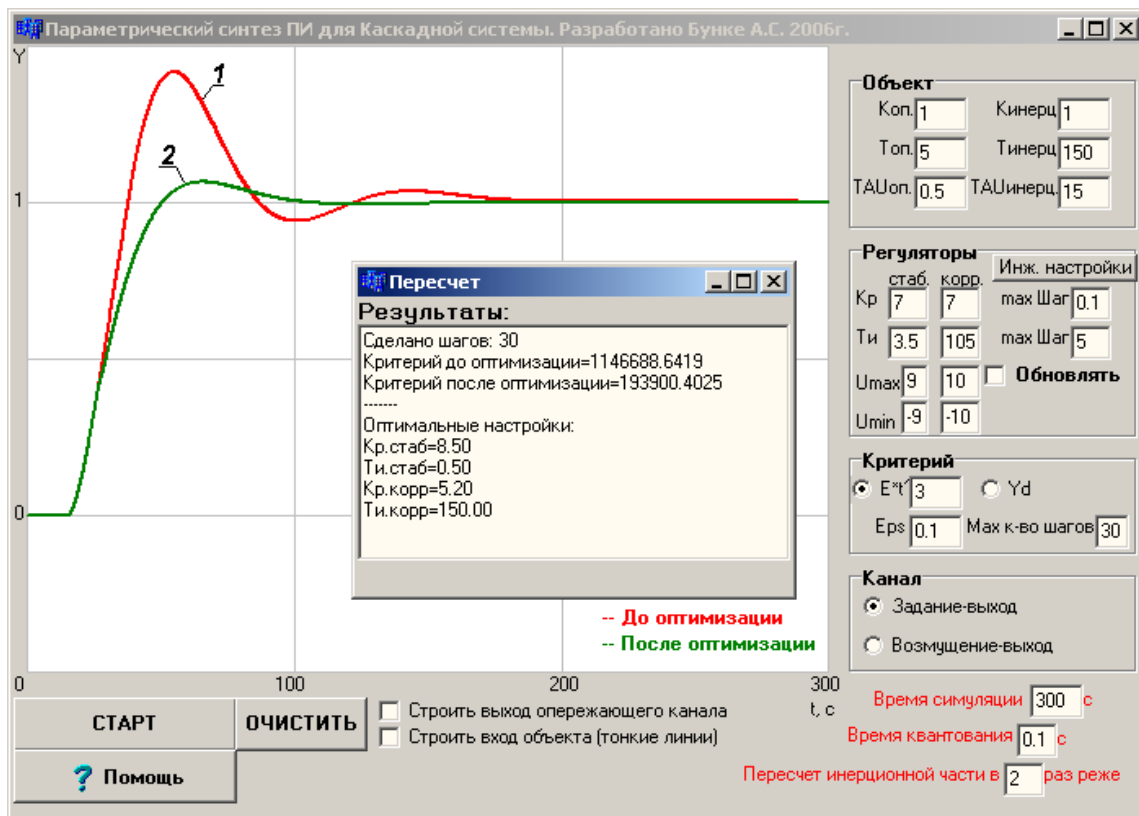


Рис. 5.16. Розрахунок каскадної АСР методом моделювання і оптимізації.
1 – перехідний процес до оптимізації; 2 – перехідний процес після 30 кроків оптимізації.

5.5. Методи випадкового пошуку

Всі розглянуті дотепер методи відносились до регулярних методів пошуку екстремуму, тобто до методів, у яких напрямок і величина робочого кроку вибираються не випадковим чином. Це означає, що при тих самих умовах задачі оптимізації, визначених алгоритмом, щоразу буде обрано один і той же напрямок пошуку та величина робочого кроку.

Існують так звані методи випадкового пошуку, що відрізняються від регулярних (детермінованих) методів оптимізації навмисним введенням

елементів випадковості. Завдяки цьому, у тих самих умовах можуть бути обрані різні напрямки і величини робочого кроку в процесі пошуку екстремуму.

Хоча методи випадкового пошуку можуть бути формально віднесені до методів нульового порядку, але наявність елементів випадковості (випадкового напрямку й випадкової точки) дозволяє виділити їх у самостійну групу.

5.5.1. Послідовність випадкових чисел

Елементи випадковості в методах випадкового пошуку реалізуються за допомогою випадкових чисел, рівномірно розподілених на деякому числовому відрізку.

Найбільш простим способом отримання випадкових чисел є вибірка їх зі спеціальних таблиць. Однак введення більших масивів випадкових чисел в ЕОМ приводить до зайвого завантаження пам'яті обчислювальної машини.

Тому найбільше поширення знайшли програмні способи одержання послідовностей випадкових чисел, засновані на використанні певних алгоритмів. Необхідно відзначити, що за допомогою цих алгоритмів можуть бути знайдені лише послідовності псевдовипадкових чисел, які насправді не є випадковими, а можуть розглядатися такими лише в тому випадку, якщо об'єм вибірки випадкових чисел не занадто великий.

Звичайно для програмних способів отримання випадкових чисел використовуються наступні алгоритми, засновані на застосуванні рекурентних співвідношень.

Алгоритм I

$$Z_{i+1} = K \cdot Z_i \bmod M$$

Це означає, що кожне наступне випадкове число Z_{i+1} дорівнює остачі від ділення на M добутку числа Z_i на постійний множник K . Рекомендується вибрати $K = 5^{17}$, $M = 2^{42}$, $Z_0 = 1$.

Алгоритм II

$$Z_{i+1} = (K_1 \cdot Z_i + K_2) \bmod K_3$$

Це означає, що кожне наступне випадкове число Z_{i+1} дорівнює остачі від ділення величини $(K_1 \cdot Z_i + K_2)$ на K_3 . Рекомендується вибрати $K_1 = 13077$, $K_2 = 6925$, $K_3 = 32767$, $Z_0 = 13$. Для знаходження випадкового напрямку \vec{a} за допомогою послідовності випадкових чисел Z_i , $i = 1, \dots, m$ необхідно проноrmувати кожне з m випадкових чисел:

$$a_i = \frac{z_i}{\sqrt{\sum_{i=1}^m z_i^2}}, \quad i = 1, \dots, m. \quad (5.16)$$

Звідси випливає, що:

$$\sum_{i=1}^m a_i^2 = 1, \quad (5.17)$$

тому випадковий вектор \vec{a} має фактично довжину, рівну одиниці, і характеризує випадковий напрямок в m -мірному просторі.

Координати випадкової точки в m -мірному просторі $u_i, i = 1, \dots, m$ також можуть бути знайдені за допомогою сукупності випадкових чисел $Z_i, i = 1, \dots, m$ рівномірно розподіленої на якому-небудь інтервалі.

Нехай випадкові числа Z_i рівномірно розподілені в інтервалі $[-c, +c]$, а m -мірний допустимий простір для u_i задається умовами:

$$\alpha \leq u_i \leq \beta, \quad i = 1, \dots, m, \quad (5.18)$$

тоді:

$$u_i = \frac{z_i(\beta - \alpha) + c \cdot (\beta + \alpha)}{2 \cdot c}, \quad i = 1, \dots, m. \quad (5.19)$$

Нехай випадкові числа z_i рівномірно розподілені в інтервалі $[\alpha, \beta]$, а m -мірний допустимий простір для u_i задається умовами: $-c \leq u_i \leq c, i = 1, \dots, m$.

Тоді:

$$u_i = \frac{(2 \cdot z_i - \alpha - \beta) \cdot c}{\beta - \alpha} \quad i = 1, \dots, m \quad (5.20)$$

Наведені співвідношення (5.19) і (5.20) придатні для перетворення рівномірно розподілених випадкових чисел з одного інтервалу в будь-який інший.

5.5.2. Сліпий пошук

Метод заснований на визначенні величини цільової функції в ряді випадкових точок \vec{u}_k , обраних у просторі незалежних змінних.

Спочатку випадковим чином вибирається початкова точка \vec{u}_1 , у якій обчислюється значення цільової функції $f_0(\vec{u}_1)$. Далі вибирається \vec{u}_2 й розраховується $f_0(\vec{u}_2)$. Якщо $f_0(\vec{u}_2)$ більше (при пошуку максимуму) $f_0(\vec{u}_1)$, то значення \vec{u}_2 й $f_0(\vec{u}_2)$ запам'ятовуються. Потім вибирається точка \vec{u}_3 , обчислюється $f_0(\vec{u}_3)$, порівнюється з $f_0(\vec{u}_2)$ і т. д.

При досить великій кількості досліджених точок можна досягти високого ступеню точності у визначенні екстремуму.

Для зменшення числа досліджуваних точок метод сліпого пошуку можна модифікувати одним з наступних способів.

По-перше, заздалегідь обмежується число досліджуваних випадкових точок, після чого краща точка приймається в якості початкової для будь-якого іншого, у тому числі й детермінованого, алгоритму.

По-друге, простір розв'язків звужується симетрично щодо кращої із заздалегідь обмеженого числа досліджуваної точки, і надалі сліпий пошук триває в цьому значно меншому просторі.

5.5.3. Метод випадкових напрямків

Алгоритм цього методу полягає в тому, що із точки \vec{u}_k , у якій відоме значення $f_0(\vec{u}_k)$ здійснюється крок у випадковому напрямку \vec{a}_k . Величина кроку визначається величиною крокового множника h_k .

У результаті знаходиться нова точка:

$$\vec{u}_{k+1} = \vec{u}_k + \vec{h}_k \cdot \vec{a}_k \quad (5.21)$$

Якщо при пошуку максимуму $f_0(\vec{u}_{k+1}) > f_0(\vec{u}_k)$, то робиться новий крок із точки \vec{u}_{k+1} у випадковому напрямку \vec{a}_{k+1} і т.д.

Якщо ж $f_0(\vec{u}_{k+1}) < f_0(\vec{u}_k)$, то вибирається наступний випадковий напрямок \vec{a}_k і робиться повторний крок із точки \vec{u}_k .

Пошук закінчується, якщо після виконання серії кроків більшого значення цільової функції знайти не вдається.

Величина крокового множника h_k може бути постійною, або змінною і вибиратися як за допомогою регулярних алгоритмів, так і за допомогою послідовностей випадкових чисел.

Існує модифікація методу випадкових напрямків – метод випадкових напрямків із зворотним кроком. Помітною її особливістю є те, що при невдалому кроці $h_k \cdot \vec{a}_k$ із точки \vec{u}_k відразу здійснюється крок у зворотному напрямку – $h_k \cdot \vec{a}_k$, тобто реалізується алгоритм:

$$\vec{u}_{k+1} = \vec{u}_k + \vec{h}_k \cdot \vec{a}_k \cdot \text{sign} \left[f_0(\vec{u}_{1_{k+1}}) - f_0(\vec{u}_{2_{k+1}}) \right] \quad (5.22)$$

Алгоритм (5.22) при достатньому віддаленні від екстремуму виявляється досить ефективним. Інша модифікація методу випадкових напрямків називається методом випадкових напрямків з лінійним прогнозом. Особливість цієї модифікації полягає в тому, що після невдалого кроку із точки \vec{u}_k робиться крок у протилежному напрямку, як у методі зі зворотним кроком, але значення цільової функції в точці $\vec{u}_{k+1} = \vec{u}_k - h_k \cdot \vec{a}_k$ не визначається, а перераховується в припущенні про лінійність цільової функції за формулою:

$$f_0(\vec{u}_{k+1}) = 2 \cdot f_0(\vec{u}_k) - f_0(\vec{u}_k + h_k \cdot \vec{a}_k) \quad (5.23)$$

Ще одна модифікація методу випадкових напрямків називається *методом найкращої проби*. У цій модифікації генерують деяке число m випадкових векторів $\{\vec{a}_k^S\} = \{\vec{a}_k^1, \vec{a}_k^2, \dots, \vec{a}_k^m\}$, $S=1, \dots, m$.

Потім здійснюється m пробних кроків за алгоритмом

$$\vec{u}_{k+1}^S = \vec{u}_k + h_k \cdot \vec{a}_k^S, \quad S=1, \dots, m, \quad (5.24)$$

і як робочий крок вибирається вектор \vec{a}_k^S , що приводить до найбільшої зміни цільової функції. Точка \vec{u}_{k+1}^S приймається в якості початкової, і процес пошуку повторюється спочатку. Очевидно, що при $m \rightarrow \infty$ процес пошуку аналогічний градієнтному методу.

5.6. Методи оптимізації при наявності обмежень

Крім методу множників Лагранжа, призначеного для розв'язку задач оптимізації з обмеженнями типу рівностей, існує два основних підходи до розв'язку задач умовної оптимізації.

Перший полягає в тому, що задача на екстремум при наявності обмежень зводиться до задачі безумовної оптимізації за допомогою спеціальних, так званих *штрафних функцій*. Методи, що використовують такі функції, називають методами штрафних функцій.

Другий підхід полягає в тому, що рух до екстремуму здійснюється по послідовності припустимих точок, тобто точок, що задовольняють обмеженням задачі з монотонно спадаючими (зростаючими) значеннями цільової функції. Цей підхід ілюструє метод проекції градієнта.

5.6.1. Методи штрафних функцій

Ці методи можуть використовуватися як при лінійних, так і при нелінійних обмеженнях типу рівностей і нерівностей.

Зміст методів полягає в тому, що вихідна цільова функція $f_0(\vec{u})$ розширюється за рахунок введення додаткових коефіцієнтів і функцій, що враховують вплив обмежень таким чином, що при порушенні обмежень різко зростає (спадає) значення розширеної цільової функції $I_p(\vec{u})$. Тим самим рішення задачі умовної оптимізації $f_0(\vec{u})$ зводиться до розв'язку задачі безумовної оптимізації $I_p(\vec{u})$.

Математична постановка задачі умовної оптимізації:

$$f_o(\bar{u}) \rightarrow \min_{\bar{u} \in V_u}, V_u = \left\{ \bar{u} \in R^n \mid g_j(\bar{u}) \geq 0, j=\overline{1,m}; h_i(\bar{u}) = 0, i=\overline{m+1,s} \right\} \quad (5.25)$$

де $f_o(\bar{u})$ - цільова функція;

V_u - область, на якій може приймати значення вектор \bar{u} ;

$g_j(\bar{u}) \geq 0, j=\overline{1,m}$ - умови типу нерівності;

$h_i(\bar{u}) = 0, i=\overline{m+1,s}$ - умови типу рівності.

Припускається, що для вектора \bar{u}^* , який є рішенням цієї задачі, відоме деяке початкове наближення $\bar{u}^{(0)}$, можливо неприпустиме, тобто таке, що не задовольняє умовам типів рівності та нерівності задачі. За допомогою алгоритмів, які розглядаються нижче, у просторі R^n будується кінцева послідовність точок $\bar{u}^{(i)}, i=\overline{1,N}$, яка починається із заданої точки $\bar{u}^{(0)}$ та завершується точкою $\bar{u}^{(N)}$, яка дає найкраще наближення до \bar{u}^* серед усіх точок побудованої послідовності. У якості $\bar{u}^{(i)}, i=\overline{1,N}$ беруться стаціонарні точки так званої штрафної функції – цільової функції допоміжної задачі безумовної мінімізації. За допомогою штрафної функції вихідна задача умовної мінімізації перетворюється у послідовність задач безумовної мінімізації. Конкретні методи, побудовані на вказаній загальній схемі, визначаються видом штрафної функції, а також правилами, за якими проводиться перерахунок штрафних параметрів по закінченню чергового циклу безумовної мінімізації. Штрафна функція, яка дозволяє обмежитися рішенням лише одної задачі безумовної мінімізації, називається точною.

Методи штрафних функцій мають просту реалізацію. Важлива їх якість також у тому, що вони дають досліднику багатий матеріал безумовної оптимізації.

Штрафна функція визначається виразом

$$F(\bar{u}) = f(\bar{u}) + P(r, g(\bar{u}), h(\bar{u})) \quad (5.26)$$

де r - набір штрафних параметрів.

Функція $P(r, g(\bar{u}), h(\bar{u}))$ називається штрафом.

Розрізняють два методи штрафних функцій – метод внутрішньої точки (бар'єрних точок) та метод зовнішньої точки.

Метод внутрішньої точки (бар'єрних точок)

Розглянемо графічну інтерпретацію задачі на прикладі.

Приклад: $f(u) = u \rightarrow \min_u, u - 2 \leq 0$.

Графічно задачу зображено на рис. 5.17.

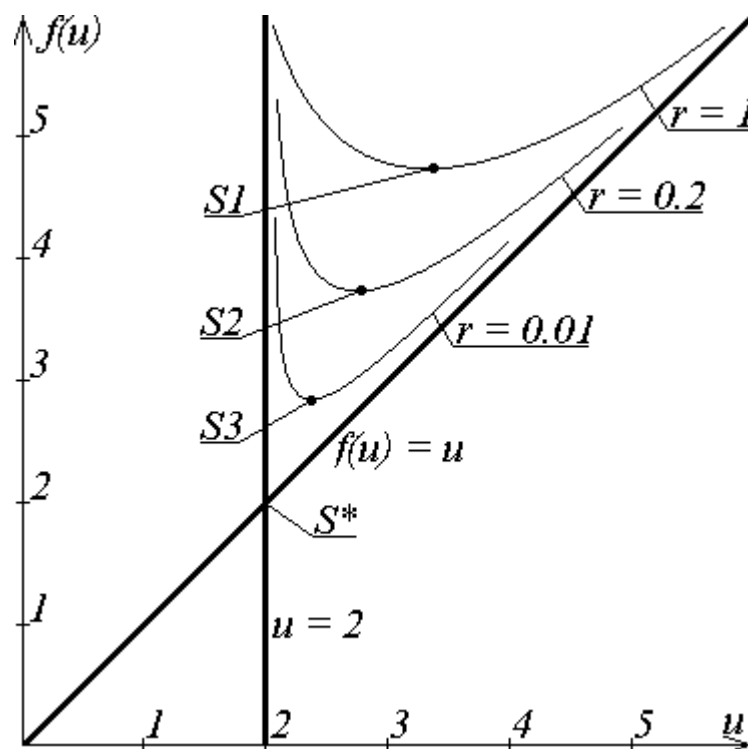


Рис. 5.17. Графічне зображення задачі умовної оптимізації

Складемо штрафну функцію: $F(u) = u + \frac{r}{u-2}$. Лінії рівня для $r = 1; 0,2; 0,01$ зображені на рис. 5.17.

Легко бачити, що послідовність точок S_1, S_2, S_3 прямують до точки S^* - мінімуму функції за наявності обмежень.

Таким чином, задача з обмеженнями перетворюється у послідовність підзадач безумовної оптимізації.

При розв'язуванні задачі необхідно обрати початкове значення штрафного параметра r та змінювати його після кожної під задачі безумовної оптимізації так, щоб забезпечити збіжність послідовності стаціонарних точок $\bar{u}^{(k)}$ до оптимального значення.

Недоліки метода:

1. Необхідність пошуку допустимої точки, щоб використати її у якості початкової.

2. Штрафна функція $F(\bar{u})$ має чітко виражену «яружну» структуру при малих r_k , що ускладнює рішення задач безумовної оптимізації $F_k(\bar{u})$.

3. Метод призначений для рішення задач нелінійного програмування з обмеженнями типу нерівності.

Метод передбачає побудову штрафів таким чином, щоб при наближенні вектора $\bar{u} \in V_u$ до границі області V_u величина $P(\bar{u}, r)$ необмежено зростає.

Траєкторія пошуку мінімуму, якщо пошук почався з внутрішньої точки, повністю буде лежати у V_u . Звідси і назва метода. Границя області V_u є своєрідним бар'єром, який неможна перейти у процесі пошуку. Звідси друга назва метода.

Для задачі

$$f(\vec{u}) \rightarrow \min_{\vec{u}}, g_i(\vec{u}) \geq 0, i=\overline{1,m}$$

у якості штрафів можуть бути побудовані наступні функції:

$$P(\vec{u}, r_k) = r_k \sum_{i=1}^m \frac{1}{g_i(\vec{u})} \quad (5.27)$$

$$P(\vec{u}, r_k) = -r_k \sum_{i=1}^m \ln g_i(\vec{u}) \quad (5.28)$$

Якщо рішення лежить на границі або біля неї, то для підвищення точності коефіцієнт r необхідно зменшувати.

Для задачі

$$f(\vec{u}) \rightarrow \min_{\vec{u}}, g_i(\vec{u}) \leq 0, i=\overline{1,m}$$

у якості штрафу може бути використано

$$P(\vec{u}, r_k) = \frac{1}{r_k} \sum_{i=1}^m (g_i(\vec{u}))^2, \quad (5.29)$$

де $g_i(\vec{u}) = \max\{0, g_i(\vec{u})\}, i=\overline{1,m}$.

Метод зовнішньої точки

Метод передбачає побудову штрафів таким чином, щоб значення перетвореної цільової функції $F(\vec{u}, r)$ у допустимій області точно або наближено дорівнювали початковій цільовій функції $f(\vec{u})$, а зовні допустимої області – значно перевищували значення $f(\vec{u})$.

Для задачі

$$f(\vec{u}) \rightarrow \min_{\vec{u} \in V_u}, \forall u = \left\{ \vec{u} \in R^m \mid h_j(\vec{u}) = 0, j=\overline{1,m} \right\}$$

можуть бути використані наступні штрафи:

$$P(\vec{u}, r_k) = \frac{1}{r_k} \sum_{j=1}^m h_j(\vec{u}); \quad (5.30)$$

$$P(\vec{u}, r_k) = \frac{1}{r_k} \sum_{j=1}^m |h_j(\vec{u})|^p, \quad (5.31)$$

де P – довільне фіксоване число.

Алгоритм оптимізації методом штрафних функцій:

1. Задання вихідних величин.
2. Побудувати $F(\vec{u}, r_k)$.

3. Знайти $\vec{u}^{(k)} = \arg \min_{\vec{u}} F(\vec{u}, r_k)$ при фіксованому r_k з використанням методу безумовної оптимізації.

4. Перевірити умову

$$\left| F(\vec{u}^{(k+1)}, r_k) - F(\vec{u}^{(k)}, r_k) \right| \leq \varepsilon_3,$$

а також, можливо

$$\begin{aligned} \|\vec{u}^{k+1} - \vec{u}^k\| &< \varepsilon_4, \\ |f(\vec{u}^{k+1}) - f(\vec{u}^k)| &< \varepsilon_5, \\ |r_{k+1} - r_k| &< \varepsilon_6. \end{aligned}$$

Умови виконуються: Кінець.

Умови не виконуються: Перехід до наступного кроку.

5. $r_{k+1} = r_k + \Delta r_k$. Перехід до кроку 2.

Комбіновані алгоритми методу штрафних функцій

Перетворена цільова функція може бути представлена у наступному вигляді:

$$F(\vec{u}) = f(\vec{u}) - r_k \sum_{j=1}^{m1} \ln g_j(\vec{u}) + \frac{1}{r_k} \sum_{i=1}^{m2} h_i^2(\vec{u}) \quad (5.32)$$

$$F(\vec{u}) = f(\vec{u}) - r_k \sum_{j=1}^{m1} \frac{1}{g_j(\vec{u})} + \frac{1}{r_k} \sum_{i=1}^{m2} h_i^2(\vec{u}) \quad (5.33)$$

5.6.2. Метод проекції градієнта

Цей метод придатний для розв'язку задач оптимізації при наявності лінійних обмежень типу рівностей і нерівностей.

Необхідно відзначити, що нелінійні обмеження можуть бути апроксимовані в околиці розглянутої точки за допомогою заміни нелінійних функцій членами першого порядку у відповідних розкладах у ряд Тейлора. Завдяки такій лінійній апроксимації, метод проекції градієнта може бути використаний і у випадку нелінійних обмежень.

Ідея методу полягає в тому, що робиться крок по проекції градієнта $\nabla f_0(\vec{u}_k)$ на множину, що задається лінійними (або лінеарізованими) активними обмеженнями. Під *активними обмеженнями* $g_j(\vec{u})$ розуміються всі обмеження типу рівностей й обмеження типу нерівностей, які в точці \vec{u}_k приймають значення рівні нулю і є лінійно незалежними. По суті, цей метод являє собою градієнтний метод у сполученні з ортогональним проектуванням антиградієнту на допустиму множину.

Алгоритм методу має вигляд:

$$\vec{u}_{k+1} = \vec{u}_k - \vec{\omega}_k, \quad (5.34)$$

$$\vec{\omega}_k = h_k \cdot \mathbf{P}_g^k \cdot \nabla f_0(\vec{u}_k), \quad (5.35)$$

$$\mathbf{P}_g^k = \mathbf{I} - \mathbf{A}_g^T (\mathbf{A}_g \cdot \mathbf{A}_g^T)^{-1} \cdot \mathbf{A}_g \quad (5.36)$$

де h_k – кроковий множник, що визначає довжину кроку в напрямку $\mathbf{P}_g^k \cdot \nabla f_0(\vec{u}_k)$. Величина його вибирається зазвичай з умов, що $f_0(\vec{u})$ повинна зменшуватися (збільшуватися) уздовж обраного напрямку, і що траєкторія руху повинна залишатися усередині допустимої області.

\mathbf{A}_g – матриця (розміром $m \times l$) частинних похідних активних лінійно незалежних обмежень, строки якої являються множиною нормалей до поверхонь обмежень:

$$\mathbf{A}_g = \begin{bmatrix} \frac{\partial g_1(\vec{u}_k)}{\partial u_1} & \dots & \frac{\partial g_1(\vec{u}_k)}{\partial u_m} \\ \dots & \dots & \dots \\ \frac{\partial g_l(\vec{u}_k)}{\partial u_1} & \dots & \frac{\partial g_l(\vec{u}_k)}{\partial u_m} \end{bmatrix}, \quad (5.37)$$

\mathbf{P}_g^k – матриця проектування градієнту $\nabla f_0(\vec{u}_k)$ на перетин активних лінійно незалежних обмежень, що задовольняють умові: $g_j(\vec{u}) = \vec{a}_j^T \cdot \vec{u} = 0$, $j = 1, \dots, l$;

Дію алгоритму (5.34)÷(5.36) необхідно продовжувати доти, доки не буде виконана умова $\mathbf{P}_g^k \cdot \nabla f_0(\vec{u}_k) = 0$, за якої точка \vec{u}_k задовольняє необхідним умовам Лагранжа (на практиці перевіряють виконання умови $\|\mathbf{P}_g^k \cdot \nabla f_0(\vec{u}_k)\| \leq \varepsilon$). Після цього обчислюється l -мірний вектор:

$$\vec{\xi}_k = (\mathbf{A}_g \cdot \mathbf{A}_g^T)^{-1} \cdot \mathbf{A}_g \cdot \vec{u}_k \quad (5.38)$$

Якщо кожна складова ξ_j , $j = 1, \dots, l$ цього вектору більше або дорівнює нулю, то це означає, що \vec{u}_k є оптимальною точкою. Якщо деякі $\xi_j < 0$, то із різноманіття обмежень необхідно виключити відповідні j -і обмеження і повторити алгоритм (5.34)÷(5.36). Останні умови базуються на тому, що якщо $\mathbf{P}_g^k \cdot \nabla f_0(\vec{u}_k) = 0$, то можна записати $\nabla f_0(\vec{u}_k) = \sum_{j=1}^l \xi_j \cdot \vec{a}_j$, де ξ_j – скалярні коефіцієнти, що складають вектор $\vec{\xi}_k$. В силу цього, якщо $\xi_j \geq 0$ для всіх $j = 1, \dots, l$, то задовольняється умова теореми Куна-Такера і, отже, \vec{u}_k є оптимальною точкою.

Метод проекції градієнту вважається досить ефективним. Його графічна інтерпретація наведена на рис. 5.18.

Основний алгоритм методу проекції градієнту:

Вирахувати P_g^k вважаючи, що всу вектори \vec{a}_j лінійно незалежні, задана допустима точка $u^{(t)}$ та допустима похибка зходимості $\varepsilon > 0$.

- 1) Вирахувати $s^{(k)} = -P_g^k \cdot \nabla f_0(\vec{u}_k)$.
- 2) Якщо $\|s^{(k)}\| \leq \varepsilon$, то обчислити $\tilde{\xi}_k$, використовуючи формулу (5.38), та завершити обчислення. В іншому випадку продовжити обчислення.
- 3) Визначити максимальну довжину кроку:

$$h_{\max} = \min \left\{ \max \left(0, \frac{b_j - a_j^T u^{(k)}}{a_j^T s^{(k)}} \text{ або } \infty, \text{ якщо } a_j^T s^{(k)} = 0 \right); j = 1, \dots, K \right\}.$$

- 4) Вирішити задачу одномірного пошуку:

мінімізувати $f(u^{(k)} + h_k \cdot s^{(k)})$, $0 \leq h \leq h_{\max}$.

- 5) Прийняти $u^{(k+1)} = u^{(k)} + h_k s^{(k)}$ та перейти до п. 1.

Для обчислення P_g^k необхідно, щоб векторі коефіцієнтів \vec{a}_j в обмеженнях були лінійно незалежними. Оскільки залежні лінійні обмеження не впливають на рішення і їх можна виключити із розгляду, ця вимога не є суттєвою.

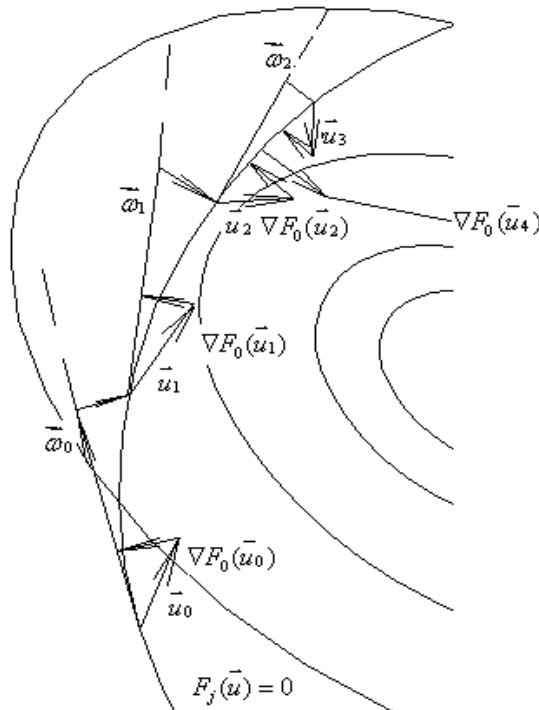


Рис. 5.18. Оптимізація методом проекцій градієнта

Приклад 5.4. Розв'язок задачі мінімізації методом проекції градієнту.

Мінімізувати $f(x) = (x_1 - 1)^2 + (x_2 - 2)^2$

При обмеженнях $g_1(x) = x_1 - 2x_2 \geq -2$

$$g_2(x) = -x_1 - x_2 \geq -4$$

$$g_3(x) = x_1 \geq 0, \quad g_4(x) = x_2 \geq 0,$$

де в якості допустимої початкової точки використовується точка $x^{(0)} = (0, 0)$.

Розв'язок.

В точці $x^{(0)}$ обмеження $g_1 \geq 0$ та $g_2 \geq 0$ неактивні, а умови невід'ємності $g_3 \geq 0$ та $g_4 \geq 0$ обертаються у рівності. Таким чином,

$$a_3 = (1; 0)^T, \quad a_4 = (0; 1)^T \quad \text{та} \quad A = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}.$$

Маємо $AA^T = I$, $(AA^T)^{-1} = I$ та $P = (I - I) = 0$.

Крок 1. $s^{(0)} = -P \cdot \nabla f = -0 \begin{pmatrix} -2 \\ -4 \end{pmatrix} = 0$. Таким чином, необхідно перевірити відповідні обмеження та множники

$$\xi^{(0)} = (AA^T)^{-1} A \nabla f = I \begin{pmatrix} -2 \\ -4 \end{pmatrix} = \begin{pmatrix} -2 \\ -4 \end{pmatrix}.$$

Оскільки $\xi_2^{(0)}$ - найменший від'ємний множник, друге активне обмеження $g_4(x) \geq 0$ виключається з множини активних обмежень. При цьому активна множина скорочується до єдиного обмеження $g_3(x) \geq 0$. Таким чином,

$$A = (1; 0) \quad \text{та} \quad AA^T = (1; 0) \begin{pmatrix} 1 \\ 0 \end{pmatrix} = 1.$$

Отже,

$$P = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} - \begin{pmatrix} 1 \\ 0 \end{pmatrix} (1; 0) = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}.$$

Крок 1.

$$s^{(0)} = -P \cdot \nabla f = -\begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} -2 \\ -4 \end{pmatrix} = \begin{pmatrix} 0 \\ 4 \end{pmatrix}.$$

Крок 2. Так як $\|s^{(0)}\| = 4 > 0$, то продовжуємо, починаючи з кроку 3.

Крок 3.

$$h_{\max} = \min \left\{ \begin{array}{l} \max \left(0, \frac{-2-0}{-8} \right) \text{ для } g_1 \\ \max \left(0, \frac{-4-0}{-4} \right) \text{ для } g_2 \\ \max (0, \infty) \text{ для } g_3, \text{ оскільки } (1, 0) \begin{pmatrix} 0 \\ 4 \end{pmatrix} = 0 \\ \max \left(0, \frac{-0-0}{4} \right) \text{ для } g_4 \end{array} \right\} = \frac{1}{4}.$$

Крок 4. Шукаємо мінімум вздовж прямої

$$x = \begin{pmatrix} 0 \\ 0 \end{pmatrix} + h \begin{pmatrix} 0 \\ 4 \end{pmatrix} \text{ при } 0 \leq h \leq \frac{1}{4}.$$

Функція $f(h) = (-1)^2 + (4h-2)^2$ монотонно спадає по h , доки значення h не стане рівним $\frac{1}{2}$. Отже,

$$h^* = h_{\max} = \frac{1}{4}.$$

Крок 5. $x^{(1)} = (0, 1)^T$.

В цій новій точці знову перевіряємо обмеження на активність. Очевидно, що обмеження $g_1 \geq 0$ та $g_3 \geq 0$ активні. Таким чином,

$$A = \begin{pmatrix} 1 & -2 \\ 1 & 0 \end{pmatrix}.$$

Легко перевірити, що $P = 0$, $s = -P \cdot \nabla f = 0$. Вектор множників $\xi^{(1)}$ дорівнює $(1, -3)^T$. Так як $\xi_2^{(1)} < 0$, друге обмеження $g_3 \geq 0$ необхідно виключити з множини активних обмежень. Тепер активна множина обмежень складається лише з $g_1 \geq 0$. Перераховуючи P при $A = (1, -2)$, одержуємо

$$P = \begin{pmatrix} 0,8 & 0,4 \\ 0,4 & 0,2 \end{pmatrix}.$$

Крок 1.

$$s^{(1)} = -P \cdot \nabla f = - \begin{pmatrix} 0,8 & 0,4 \\ 0,4 & 0,2 \end{pmatrix} \begin{pmatrix} -2 \\ -2 \end{pmatrix} = \begin{pmatrix} 2,4 \\ 1,2 \end{pmatrix}.$$

Крок 2. $\|s^{(1)}\| > 0$; тому продовжуємо обчислення.

Крок 3. Довжина кроку $h_{\max} = \frac{5}{6}$ визначається обмеженням $g_2 \geq 0$.

Крок 4. Далі шукаємо мінімум функції

$$f(h) = (2,4h-1)^2 + (1+1,2h-2)^2.$$

Мінімум досягається в $h^* = 1/2 < h_{\max}$.

Крок 5. Новою точкою є $x^{(2)} = (1, 2; 1, 6)^T$. В цій точці активне лише перше обмеження. Множина активних обмежень лишається без змін, тому немає необхідності перераховувати P . В точці $x^{(2)}$ маємо

$$s^{(1)} = -P \cdot \nabla f = - \begin{pmatrix} 0,8 & 0,4 \\ 0,4 & 0,2 \end{pmatrix} \begin{pmatrix} 0,4 \\ -0,8 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix},$$

$$\xi^{(2)} = (AA^T)^{-1} A \nabla f = 1/5 (1; -2) \begin{pmatrix} 0,4 \\ -0,8 \end{pmatrix} = 0,4 > 0.$$

Таким чином, точка $x^{(2)}$ задовольняє умовам Куна-Такера.

5.7. Контрольні завдання

1. Знайти методом Гауса-Зейделя мінімум функції

$$f(x) = (x_1 - x_2)^2 + \frac{(x_1 + x_2 - 10)^2}{9}. \text{ Початкова точка } x^0 = (0; 1).$$

2. Надаючи параметрам α і β в методі Розенброка різні значення, визначте найбільш ефективні з них для випадка квадратичної функції

$$f(x) = 10x_1^2 + 0,1x_2^2.$$

Повторіть те саме для

$$f(x) = 0,1x_1^2 + 10x_2^2.$$

3. Використайте пошук Розенброка для знаходження мінімуму цільової функції

$$f(x) = 3x_1^2 + x_2^2$$

при $\alpha = 3$ і $\beta = \frac{1}{2}$, доки подальше покращення не стане неможливим.

4. Задана функція має мінімуми в точках $(0; 1)$ та $(1; 0)$ та сідлову точку $(4^{-1/3}; 4^{-1/3})$. Починаючи з точки $x = (3; 3)$, побудуйте траєкторії мінімізації (вказуючи значення цільової функції $f(x) = (x_1^2 + x_2^2 - 1)^2 + (x_1 + x_2 - 1)^2$ в залежності від числа ітерацій) для методів:

- а) Хука-Дживса
- б) пошук по симплексу
- в) Розенброка

Що станеться, якщо почати з точки $(4^{-1/3}; 4^{-1/3})$?

Мінімізуйте задану функцію методами випадкового пошуку.

5. Знайдіть мінімум та максимум функції

$$f(x) = 20 + 0,3x_1 - 4x_2 + 0,3x_1^2 + 0,3x_2^2 + 0,4x_1x_2$$

одним з методів пошуку. Починайте з точок:

- а) $x^0 = (0,25; 2,5)$,
- б) $x^0 = (2,5; 2,5)$,
- в) $x^0 = (-0,25; -2,5)$.

6. Мінімізувати функцію

$$f(x) = (x_1 - 4)^2 + (x_2 - 4)^2$$

методом штрафних функцій при обмеженні $h(x) = x_1 + x_2 - 5 = 0$.

7. Мінімізувати функцію

$$f(x) = (x_1 - 4)^2 + (x_2 - 4)^2$$

методом штрафних функцій при обмеженні $g(x) = 5 - x_1 - x_2 \geq 0$.

8. Дана задача:

Мінімізувати функцію $f(x) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2$

при обмеженнях:

$$g_1(x) = x_1 + 1 \geq 0;$$

$$g_2(x) = 1 - x_2 \geq 0;$$

$$g_3(x) = 4x_2 - x_1 - 1 \geq 0;$$

$$g_4(x) = 1 - 0,5x_1 - x_2 \geq 0.$$

Запишіть для цієї задачі штрафні функції, використовуючи:

1) логарифмічний штраф;

2) штраф, що задається оберненою функцією. Дослідіть кожну з функцій в точках $x^{(0)} = (-1; 1)$ та $x^{(0)} = (-0,5; 0,5)$, які складності при цьому виникають?

9. Мінімізувати методом проекції градієнта функцію

$$f(x) = x_1^2 + x_2^2$$

при обмеженні $h_1(x) = (x_1 - 1)^3 - x_2^2 = 0$.

10. Мінімізувати методом проекції градієнта функцію

$$f(x) = x_1^2 + 2x_2^2$$

при обмеженнях:

$$g_1(x) = x_1^2 + x_2^2 \leq 0;$$

$$h_2(x) = 2x_1 - 2x_2 - 1 = 0.$$

11. Мінімізувати методом проекції градієнта функцію

$$f(x) = 4x_1 - x_2^2 - 12$$

при обмеженнях:

$$h_1(x) = 25 - x_1^2 - x_2^2 = 0;$$

$$g_2(x) = 10x_1 - x_1^2 + 10x_2 - x_2^2 - 34 \geq 0;$$

$$g_3(x) = x_1 \geq 0;$$

$$g_4(x) = x_2 \geq 0.$$

6. ЛІНІЙНЕ ПРОГРАМУВАННЯ

6.1. Постановка задачі

Лінійне програмування широко використовується при розв'язку економічних задач, задач планування виробництва, оперативного керування виробничим процесом та ін.

Часто виявляється доцільним зведення до лінійних нелінійних постановок задач, щоб використати добре розвинутий математичний апарат лінійного програмування.

Задача лінійного програмування (ЛП) формулюється в такий спосіб: необхідно визначити максимум (мінімум) лінійної функції мети, незалежні змінні якої повинні задовольняти ряду обмежень типу рівностей і нерівностей. Математичне формулювання задачі має вигляд:

$$I = f_0(x, u) = \sum_{i=1}^m c_i \cdot u_i \rightarrow \max, \quad (6.1)$$

$$\sum_{i=1}^m a_{ij} u_i \leq b_j, \quad j = 1, 2, \dots, m_1, \quad (6.2)$$

$$\sum_{i=1}^m a_{ij} u_i \geq b_j, \quad j = m_1 + 1, \dots, m_2, \quad (6.3)$$

$$\sum_{i=1}^m a_{ij} u_i = b_j, \quad j = m_2 + 1, \dots, n, \quad (6.4)$$

$$u_i \geq 0, \quad i = \overline{1, m}. \quad (6.5)$$

Кількість обмежень (6.3) може бути довільно великою, у той час як для обмежень типу рівностей справедливий наступний вираз:

$$n - m_2 < m. \quad (6.6)$$

Коефіцієнти цільової функції c_i , всі коефіцієнти a_{ji} , а також b_j повинні бути задані.

6.2. Нормальна форма задачі ЛП.

Перед тим, як вивчати принципи й алгоритми розв'язку, необхідно звести задачу до стандартного вигляду, до так званої нормальної форми.

Будь-яку лінійну нерівність (6.2) можна представити у вигляді рівності після введення додаткової змінної u_{m+1} (іноді u_{m+1} називають залишковою або надлишковою змінною):

$$\sum_{i=1}^m a_{ij} u_i + u_{m+1} = b_j, \quad u_{m+1} \geq 0. \quad (6.7)$$

Аналогічно нерівність (6.3) можна звести до рівності:

$$\sum_{i=1}^m a_{ij} u_i - u_{m+1} = b_j, u_{m+1} \geq 0. \quad (6.8)$$

Тоді лінійна задача оптимізації (6.1) ÷ (6.5) має вигляд:

$$I = f_0(\vec{u}) = c_1 \cdot u_1 + \dots + c_m \cdot u_m; \quad (6.9)$$

$$\left\{ \begin{array}{l} a_{11} \cdot u_1 + a_{12} \cdot u_2 + \dots + a_{1m} \cdot u_m + u_{m+1} = b_1; \\ \vdots \\ a_{m_1,1} \cdot u_1 + a_{m_1,2} \cdot u_2 + \dots + a_{m_1,m} \cdot u_m + u_{m+m_1} = b_{m_1}; \\ a_{m_1+1,1} \cdot u_1 + a_{m_1+1,2} \cdot u_2 + \dots + a_{m_1+1,m} \cdot u_m - u_{m+m_1+1} = b_{m_1+1}; \\ \vdots \\ a_{m_2,1} \cdot u_1 + a_{m_2,2} \cdot u_2 + \dots + a_{m_2,m} \cdot u_m - u_{m+m_2} = b_{m_2}; \\ a_{m_2+1,1} \cdot u_1 + a_{m_2+1,2} \cdot u_2 + \dots + a_{m_2+1,m} \cdot u_m - u_{m+m_2+1} = b_{m_2+1}; \\ \vdots \\ a_{n,1} \cdot u_1 + a_{n,2} \cdot u_2 + \dots + a_{n,m} \cdot u_m = b_n; \end{array} \right. \quad (6.10)$$

$$u_i = 0, i = 1, 2, \dots, m + m_2. \quad (6.11)$$

При розмірності незалежної змінної $p = m + m_2$ і виборі $c_{m+1} = \dots = c_{m+m_2} = 0$ задача в нормальній формі має вигляд:

$$I = f_0(\vec{u}) = \sum_{i=1}^p c_i u_i; \quad (6.12)$$

$$\sum_{i=1}^p a_{ji} u_i = b_j, j = \overline{1, n}; \quad (6.13)$$

$$u_i \geq 0, i = \overline{1, p}. \quad (6.14)$$

У векторній формі:

$$I = \vec{c}^T \vec{u}; \quad (6.15)$$

$$A\vec{u} = \vec{B}, \vec{u} \geq 0, \vec{B} \geq 0, \quad (6.16)$$

де A – матриця розмірності $n \times p$;

\vec{u} – вектор-стовпчик розмірністю $p \times 1$;

\vec{B} – вектор-стовпчик розмірністю $n \times 1$;

\vec{c} – вектор-стовпчик розмірністю $p \times 1$;

Зазвичай A називають *матрицею коефіцієнтів*, \vec{u} – *вектором змінних*, \vec{B} – *вектором ресурсів*, \vec{c} – *вектором оцінок* задачі ЛП.

Іноді в задачі присутні необмежені змінні, тобто змінні, які приймають негативні й позитивні значення. Тоді порушується умова (6.5). У цьому випадку варто скористатися наступною заміною. Нехай u_p – необмежена змінна. Тоді:

$$u_p = u_{p+1} - u_{p+2}, u_{p+1} \geq 0; u_{p+2} \geq 0. \quad (6.17)$$

6.3. Властивості лінійних задач оптимізації

Лінійні задачі оптимізації і їх розв'язки мають деякі характерні властивості, які цілеспрямовано можуть бути використані для формулювання умов оптимальності й алгоритмів розв'язків. Для наочного подання цих властивостей розглянемо задачу із двома змінними величинами і її геометричним трактуванням.

Приклад 6.1. Лінійна задача оптимізації

Дано лінійну задачу оптимізації:

$$f_0(\vec{u}) = u_1 + u_2 \rightarrow \max_u, \quad (6.18)$$

з обмеженнями:

$$\begin{cases} 2u_1 + 2u_2 \leq 1, \\ 2u_1 + 3u_2 \leq 2, \\ 4u_1 - 3u_2 \leq 1, \\ u_1 \geq 0, u_2 \geq 0. \end{cases} \quad (6.19)$$

Розв'язок

1) Знаходження допустимої області розв'язків. Область розв'язків визначають обмеження (6.19). Покажемо на рис. 6.1 прямі, які відповідають математичним відношенням (6.1) зі знаком рівності. Область допустимих розв'язків для розглянутої задачі має вигляд п'ятикутника ABCMD.

2) Знаходження максимуму цільової функції. Цільова функція графічно може бути представлена прямою:

$$u_1 + u_2 = I_1 = \text{const}. \quad (6.20)$$

На малюнку вона зображена для $I_1 = 0.3$. Легко побачити, що збільшення значення функції мети можливо лише в напрямку стрілки шляхом паралельного зсуву прямої I_1 . Паралельний зсув можна робити лише до точки M , тому що в цій точці досягається границя допустимої області розв'язків.

Будь-яке подальше переміщення у напрямку зростання цільової функції буде давати лише неприпустимі розв'язки. При цьому точка $M(0,25; 0,5)$ буде оптимальним розв'язком задачі. Значення цільової функції складе: $f_0^*(u) = 0.75$.

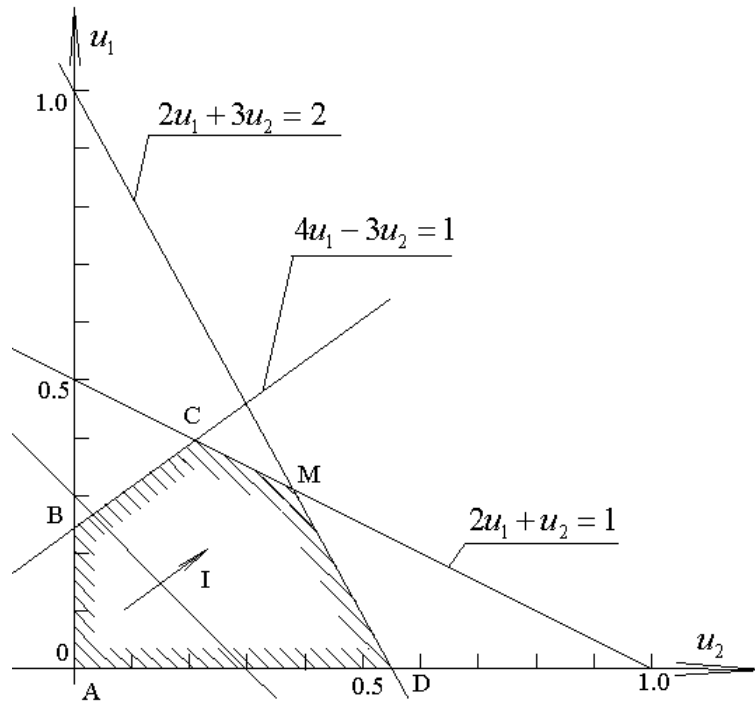


Рис. 6.1. Графічна інтерпретація лінійних обмежень

Обговорення розв'язку.

Оптимальний розв'язок знайдений в кутовій точці допустимої області. Це залишається справедливим і при збільшенні числа обмежень.

Оскільки множина допустимих розв'язків замкнута і не пуста, то в даній ситуації існує тільки один особливий випадок. Цей особливий випадок має місце, якщо пряма I_1 , що зображує цільову функцію, проходить паралельно якому-небудь обмеженню (на рис. 6.1 прямій CM). Тоді паралельне переміщення до меж допустимої області розв'язків призведе до накладання прямої I_1 на сторону CM багатокутника. При цьому задача буде мати нескінченну множину оптимальних розв'язків, включаючи обидві кутові точки. У цьому випадку всі розв'язки є рівнозначними. На практиці знання одного з оптимальних розв'язків є достатнім, тому можна обмежитися розв'язком, представленим однією із двох кутових точок.

Для деяких задач ЛП не існує оптимального розв'язку. У прикладі така ситуація виникає, якщо опустити обмеження $2u_1 + 3u_2 \leq 2$ й $2u_1 + u_2 \leq 1$. У цьому випадку допустима область не замкнута, а максимум цільової функції дорівнює $+\infty$. Говорять, що задача має необмежений оптимум.

Перед тим як робити висновки з розглянутої задачі, введемо поняття базисного рішення.

Базисний розв'язок.

В задачі ЛП, як правило, число рівнянь обмежень задачі менше числа змінних, тобто $n < p$.

Тому множина її допустимих розв'язків нескінченна. Для вибору найкращого допустимого розв'язку вихідну систему зводять до канонічного виду:

$$\begin{aligned} u_1 + \alpha_{1,n+1} \cdot u_{n+1} + \dots + \alpha_{1p} \cdot u_p &= d_1; \\ \vdots \\ u_n + \alpha_{n,n+1} \cdot u_{n+1} + \dots + \alpha_{np} \cdot u_p &= d_n. \end{aligned} \tag{6.21}$$

Для цього використовують два типи операцій:

- 1) Множення рівняння на яке-небудь число.
- 2) Множення рівняння на число й додавання його до будь-якого іншого рівняння.

Кожна змінна u_1, \dots, u_n входить тільки в одне з рівнянь (6.21) з одиничним коефіцієнтом, а в інші - з нульовими. Ці змінні називаються *базисними* або *залежними*.

У канонічній системі кожному рівнянню відповідає одна базисна змінна, а інші - незалежні.

З (6.21) легко одержати базисне рішення при нульових значеннях небазисних або незалежних змінних. У системі (6.21) $u_{n+1} = \dots = u_p = 0$, а базисне рішення має вигляд $u_1 = d_1, \dots, u_n = d_n$.

Базисний розв'язок є допустимим, якщо значення базисних змінних, які входять у нього, невід'ємні.

Кожна кутова точка допустимої області розв'язків якої-небудь задачі ЛП відповідає допустимому базисному розв'язку, так само як будь-який допустимий базисний розв'язок утворить кутову точку.

Можна довести, що не пуста множина допустимих розв'язків має, щонайменше, одну або ж багато кутових точок. При цьому наявність допустимого базисного розв'язку є гарантованим, оскільки допустима множина розв'язків не порожня. В цілому можна зробити наступні висновки щодо розв'язків задач ЛП:

1) Якщо множина допустимих розв'язків не порожня, а цільова функція обмежена зверху (задача на максимум), то існує, принаймні, один оптимальний розв'язок.

2) Якщо існує оптимальний розв'язок, то принаймні одна з кутових точок допустимої області являє собою оптимальний розв'язок.

3) Оптимальний розв'язок задачі ЛП можна знайти шляхом перебору допустимих базисних розв'язків.

6.4. Симплексний метод розв'язку задач ЛП

Для розв'язку лінійних задач оптимізації розроблений цілий ряд чисельних методів, з яких найвідомішим є симплексний. Перш, ніж докладно говорити про цей метод, необхідно згадати наступне, що наводилося раніше твердження: якщо існує оптимальний розв'язок, то він завжди перебуває в одній з кутових точок допустимої області.

На основі цього твердження розроблений алгоритм, за допомогою якого можна поступово, починаючи з якої-небудь кутової точки допустимої області розв'язків (допустимий базисний розв'язок) рухатися до наступних суміжних допустимих базисних розв'язків (ДБР) з більшим значенням цільової функції. При цьому, як початкова умова, потрібне знання якого-небудь ДБР. Якщо деяке ДБР не можна поліпшити, то воно є оптимальним, і розрахунки припиняються.

У цьому розділі без доказу розглянемо основи симплекс-методу і простий числовий приклад.

6.4.1. Основи симплексного методу

Нехай вихідна задача ЛП містить p змінних й n обмежень у формі рівностей. Відомий базисний розв'язок з n позитивними змінними, які позначимо через $u_{e1}, u_{e2}, \dots, u_{en} \dots$

Розв'язок системи обмежуючих рівнянь щодо базисних змінних u_{ei} має вигляд:

$$u_{ai} = - \sum_{j=1}^{p-n} \alpha_{ij} \cdot u_{n+j} + d_i, i = 1, 2, \dots, n. \quad (6.22)$$

Після підстановки (6.22) в (6.1) цільова функція може бути перетворена в такий спосіб:

$$I = f_0(\vec{u}) = c_0^* - \sum_{i=1}^{p-n} c_{n+j}^* \cdot u_{n+j}, \quad (6.23)$$

де:

$$c_0^* = \sum_{i=1}^n c_{ai} d_i; \quad c_{n+j}^* = \sum_{i=1}^n c_{ai} d_{ij} - c_{n+j}. \quad (6.24)$$

Розглянемо основні співвідношення симплексного методу пошуку оптимального розв'язку, які використовуються для цілеспрямованого перебору базисних розв'язків і перевірки оптимальності.

1) Перевірка оптимальності знайденого розв'язку.

Ознакою оптимальності знайденого базисного розв'язку є умова:

$$c_{n+j}^* \geq 0, j = \overline{1, p-n}, \quad (6.25)$$

тобто відносні оцінки небазисних змінних допустимого базисного розв'язку задачі на максимум повинні бути більшими або рівними нулю. Дійсно, у цьому випадку згідно (6.22) будь-яка зміна НБЗ u_{n+j} призведе до зменшення мети цільової функції (6.23), тому що всі $u_{n+j} = 0, j = \overline{1, p-n}$, а змінювати їх можна тільки у бік збільшення, тому що повинна виконуватися умова $u_i = 0, i = \overline{1, p}$.

Таким чином, суміжному допустимому базисному розв'язку буде відповідати значення цільової функції, що не перевершує вже досягнутого. Отже, у знайдений точці має місце локальний максимум, що через лінійність I , збігається із глобальним.

2) Знаходження суміжного допустимого базисного розв'язку.

Визначення суміжного базисного допустимого розв'язку із кращою цільовою функцією – одне з основних питань у симплекс-методі.

Суміжний базисний допустимий розв'язок відрізняється від попередніх тільки однією базисною змінною. Для того щоб його одержати, необхідно перевести одну з небазисних змінних у базис, а базисну змінну в небазисну, причому таким чином, щоб ця заміна давала максимальний приріст цільової функції.

3) Визначення небазисної змінної величини (НБЗ), що вводиться у базис.

Правило скалярного добутку.

Перетворення небазисної змінної в базисну призводить до збільшення її значення від нуля до якоїсь позитивної величини.

При цьому вона повинна збільшувати значення цільової функції I , причому давати їй максимальний приріст. Найбільший приріст забезпечує та НБЗ, біля якої в цільовій функції стоїть найменший коефіцієнт c_{n+j}^* (6.24). Тому вибір змінної, котру варто перевести в базис, варто робити по найменшому коефіцієнту c_{n+j}^* в цільовій функції.

Величина c_{n+j}^* називається відносною оцінкою НБЗ u_{n+j} , а вираз $c_{n+j}^* = \sum_{i=1}^n c_{\omega i} \alpha_{ij} - c_{n+j}$ називають правилом скалярного добутку.

4) Визначення базисної змінної величини (БЗВ), виведеної з базису.

Правило мінімального відношення.

Нехай для введення в базис обрана небазисна змінна u_s , котрій будемо привласнювати значення рівне одиниці. Тоді рівняння (6.22)

матимуть вигляд:

$$u_i = d_i - \alpha_{is} u_s, i = \overline{1, n}, u_{\xi} = 0, j = \overline{n+1, p}, j \neq S.$$

При збільшенні u_s від нуля до одиниці значення інших базисних змінних міняються відповідно до рівнянь (6.22).

Якщо $\alpha_{is} < 0$, то u_i зростає, а при $\alpha_{is} > 0$ u_i спадає і може стати негативною, а базисний розв'язок недопустимим.

Щоб базисний розв'язок не став недопустимим u_s не повинне перевищити $\min[d_i/\alpha_{is}]$, тобто:

$$\max u_s = \min_{\alpha_{is} > 0} d_i/\alpha_{is}. \quad (6.26)$$

Визначимо змінну u_r , для якої:

$$d_r/\alpha_{rs} = \min_{\alpha_{is} > 0} [d_i/\alpha_{is}] \quad (6.27)$$

Базисна змінна u_r при збільшенні u_s до d_r/α_{rs} першою серед базисних змінних перетворюється в нуль. Тому її і варто замінити в базисі змінною u_s .

Рівняння (6.27), за допомогою якого визначаються базисні змінні, виведені з базису, називається *правилом мінімального відношення*.

Для виконання розрахунків, відповідно до розглянутих вище крокам, вводиться таблиця наступної структури.

Таблиця 6.1

	$u_1 u_2 \dots \dots \dots$ u_p	u_0	Q	-1
u_{s1} u_{s2} \vdots u_{sn}	<i>Коефіцієнти обмежуючих рівнянь у нормальній формі</i>	d_1 d_2 \vdots d_n		
	<i>Коефіцієнти цільової функції згідно (6.1).</i>	c_0		

Подібна таблиця складається для кожної ітерації симплексного алгоритму, що у свою чергу складається з окремих кроків або операцій, результати яких заносяться до таблиці.

6.4.2. Алгоритм симплекса-методу в табличній формі

Операція 1. Перевіряємо оптимальності розв'язку.

Якщо всі коефіцієнти при небазисних змінних (НБЗ) у рядку $(n+1)$ більші нуля або дорівнюють нулю $u_{n+j}^* \geq 0$ (коефіцієнти u_{n+j}^* підкреслені), то

оптимальний розв'язок досягнутий. Значення базисних змінних величин (БЗВ) міститься в стовпчику u_0 таблиці.

$$u_{ei} = d_i, i = \overline{1, n} \quad (6.28)$$

Відповідне значення цільової функції буде дорівнювати c_0^* :

$$I = f_0(\vec{u}) = c_0^* \quad (6.29)$$

Розв'язок завершений.

Операція 2. Вибираємо нову БЗВ. Якщо в основному рядку є негативні коефіцієнти при НБЗ, то розв'язок триває. Найменше негативне число позначимо значком " \uparrow ", а відповідний стовпчик назовемо *стрілочним*. Змінна величина, що відповідає стрілочному стовпчику, є новою базисною змінною, котру варто ввести в базис, але попередньо треба вивести звідти одну з колишніх БЗВ.

Операція 3. Заповнюємо стовпчик Q таблиці. Для цього вибираємо позитивні числа стрілочного стовпчика, на які ділимо відповідні числа в стовпчику u_0 і результат заносимо в стовпчик Q:

$$Q = \frac{\text{число в стовпчику } u_0}{\text{число в стрілочному стовпчику}}. \quad (6.30)$$

Місця в стовпчику Q, які відповідають негативним значенням стрілочного стовпчика, не заповнюються.

Якщо стрілочний стовпчик не містить позитивних чисел, то задача нерозв'язна, і процес розв'язку припиняється.

Рядок, у якому знаходиться мінімальне число стовпчика Q, називається *основним рядком* і позначається знаком " \leftarrow ". Відповідна цьому рядку базисна змінна величина стає НБЗ. Число на перетині основного рядку і стрілочного стовпчика обведемо кружком.

Операція 4. У стовпець " \uparrow " заносимо числа зі стрілочного стовпця " \uparrow " із протилежним знаком. Відповідне місце в головному рядку не заповнюється.

Операція 5. Готується нова таблиця для розрахунків. У стовпчик БЗВ вносяться БЗВ нового припустимого базисного розв'язку в незмінному порядку. Нова базисна змінна ставиться на місце старої. Рядок з новою базисною змінною містить знак " \rightarrow " і називається допоміжним рядком.

Операція 6. Всі елементи основного рядка діляться на елемент, відзначений кружком, а результат заноситься відповідно в допоміжний рядок нової таблиці.

Операція 7. Кожен елемент α_{ij}^{v+1} нової таблиці розраховується за наступним правилом:

$$\alpha_{ij}^{v+1} = \alpha_{ij} + P_{ij} \quad (6.31)$$

де v – номер етапу симплексного алгоритму.

P_{ij} – добуток числа з рядка i , що знаходиться в стовпчику " $- \uparrow$ ", на число, що стоїть в стовпчику j допоміжного рядка " \rightarrow ".

Це правило справедливе також для елементів стовпчика й основного рядка.

Нова таблиця заповнюється до двох останніх стовпчиків (Q і " $- \uparrow$ "). Цим закінчується ряд операцій, що відноситься до етапу алгоритму. Потім переходимо до операції 1.

Приклад 6.2. Симплекс-метод

Цільова функція:

$$f_0(\vec{u}) = 2u_1 + 4u_2 \rightarrow \max_u,$$

Обмеження:

$$4u_1 + u_2 \leq 36;$$

$$u_1 + 3u_2 \leq 24;$$

$$u_1 + u_2 \leq 12;$$

$$u_i \geq 0, i = \overline{1,3}.$$

Обмеження наведемо в нормальній формі:

$$4u_1 + 2u_2 + u_3 = 36;$$

$$u_1 + 3u_2 + u_4 = 24;$$

$$u_1 + u_2 + u_5 = 12;$$

$$u_i \geq 0, i = \overline{1,5}.$$

Перший допустимий базисний розв'язок легко визначається при $u_1 = u_2 = 0$:

$$u_6 = [u_3, u_4, u_5]^T; u = [0, 0, 36, 24, 12]^T.$$

Згідно (5.23) записуємо цільову функцію:

$$f_0(u) = 0 - (-2u_1 - 4u_2).$$

Заповнюємо таблицю, що на першому етапі має такий вигляд:

Таблиця 3

БЗВ	u_1	u_2	u_3	u_4	u_5	u_0	Q	$- \uparrow$
u_3	4	1	1	0	0	36	36	-1
u_4	1	③	0	1	0	24	8	-3
u_5	1	1	0	0	1	12	12	-1
	-2	-4	↑	0	0	$c_0 = 0$	0	4

Задача вирішується за дві ітерації. Інші таблиці заповнюються відповідно до викладеного алгоритму і тут не наведені. В останній третій таблиці міститься оптимальний розв'язок:

$$u_1^{opt} = 6, u_2^{opt} = 6, u_3^{opt} = 6, u_4 = u_5 = 0;$$

$$f_0^{opt}(\vec{u}) = 36.$$

6.5. Розв'язок задач на мінімум

Для розв'язку задачі на мінімум можна перетворити її шляхом множення цільової функції на -1 у задачу максимізації і вирішувати по викладеному алгоритму.

З іншого боку, для розв'язку задачі мінімізації можна застосувати викладений алгоритм, змінивши операції 1 й 2. Критерієм оптимальності буде тепер від'ємність коефіцієнтів c_{n+j}^* . Нова базисна змінна вибирається по найбільшому c_{n+j}^* .

6.6. Вибір початкового допустимого базисного розв'язку

Без початкового базисного рішення (ПБР) не можна побудувати симплекс-таблицю. Для знаходження ПБР широко застосовують метод штучного базису.

Він полягає в тому, що для кожного обмеження (у нормальній формі) перевіряється існування базисної змінної, і якщо її нема, то вводиться штучна, яка задовольняє умові позитивності (6.14). В оптимальному розв'язку штучні змінні повинні обернутися на нуль.

Штучні змінні, а також базисні змінні (якщо вони існують для окремих обмежень) становлять допустимий базисний розв'язок для допоміжної задачі, що містить штучні змінні.

Знайдений допустимий базисний розв'язок допоміжної задачі неприпустимий для початкової, оскільки штучні змінні позитивні і вихідні обмежуючі рівняння цей розв'язок не задовольняють. З іншого боку, допустимий базисний розв'язок допоміжної системи, у якому штучні змінні дорівнюють нулю, допустимий і для вихідної задачі.

Тому спочатку вирішується задача мінімізації для допоміжної цільової функції, що містить лише штучні змінні. Така цільова функція повинна мати найменше значення рівне нулю. При цьому й штучні змінні обертаються в нуль. А знайдений базисний розв'язок буде оптимальним для допоміжної задачі та припустимим для вихідної. Далі вирішується основна задача звичним чином.

6.7. Приклади постановок задач ЛП

Оптимальне планування виробництва.

Нехай u_j - запланований підприємством обсяг виробництва j -ї продукції. Тоді вектор $\vec{u} = [u_1, \dots, u_m]^T$ - план виробництва. Загальний прибуток виробництва є цільовою функцією і буде дорівнювати:

$$I = f(\vec{u}) = \sum_{j=1}^m c_j u_j, \quad (6.32)$$

де c_j – прибуток від одиниці j -ї продукції.

Для виробництва j -ї одиниці продукції потрібно a_{ij} одиниць i -го ресурсу (сировини, енергії). Загальна кількість i -го ресурсу обмежена:

$$\sum_{j=1}^m a_{ij} \cdot u_j \leq b_i, \quad i = \overline{1, n}. \quad (6.33)$$

По фізичному змісті завдання (6.32) $u_j \geq 0$.

Завдання полягає в максимізації прибутку (6.32) при обмеженнях (6.33) і є завданням ЛП.

Транспортне завдання

Оптимізація транспортних проблем має велике господарське значення. Високий ступінь взаємозв'язку сучасних підприємств вимагає тісної кооперації між ними, незважаючи на різний профіль виробництва. Звідси виходять складні взаємини типу "виробник-споживач", які характеризуються багатобічним обміном сировиною, напівфабрикатами і готовими виробами. Залежно від конкретних умов при цьому можуть виникнути різноманітні завдання по оптимізації. Крім того, як транспортні інтерпретують й інші завдання оптимізації, як, наприклад, вибір місця для знову споруджуваного промислового підприємства або для знову обладнаних виробничих потужностей, а також завантаження устаткування та ін.

Нехай є m виробників і споживачів якого-небудь виробу. Його запас позначимо через a_i . Кількість потрібного споживачам продукту позначимо b_j . Витрати на перевезення одиниці продукції від i -го виробника до j -го споживача рівні c_{ij} .

Необхідно визначити таку кількість поставок u_{ij} , щоб забезпечити всіх споживачів при мінімальних транспортних витратах.

Математично це завдання формулюється в такий спосіб:

Визначити $I = \sum_{i=1}^m \sum_{j=1}^n c_{ij} \cdot u_{ij} \rightarrow \min$, при обмеженнях:

$$\sum_{j=1}^n u_{ij} \leq a_i, i = \overline{1, m}, \quad (6.34)$$

$$\sum_{i=1}^m u_{ij} = b_j, j = \overline{1, n}, \quad (6.35)$$

$$u_{ij} \geq 0. \quad (6.36)$$

Одним з варіантів є класичне транспортне завдання, що включає тільки обмеження типу (6.34) при дотриманні вимоги збігу попиту та пропозиції.

Класичне транспортне завдання відрізняється від звичайних завдань ЛП тим, що матриця коефіцієнтів містить лише нулі й одиниці. На цій основі була розвинута спеціальна теорія для розв'язку таких завдань.

Задача оптимального розподілу навантаження між паралельно працюючими енергоблоками АЕС

Змістовна постановка

Оптимізувати розподіл навантаження між паралельно працюючими енергоблоками, що забезпечуватиме мінімум питомих затрат палива всіх енергоблоків на виробництво електроенергії.

Математична постановка задачі

Задача оптимального розподілу навантаження в математичній формі може бути записана наступним чином:

$$I = \sum_{k=1}^n \frac{\beta_k}{N} \rightarrow \min$$

де k – номер енергоблоку,
 I – сумарне електричне навантаження на електростанцію,
 N – кількість енергоблоків.

Математична модель енергоблоку:

$$\beta_k(N_i) = \gamma_k \cdot a \cdot N_i^2$$

де β_k – витрата палива на i -й енергоблок,
 N_i – електричне навантаження i -го енергоблоку,
 a – коефіцієнт пропорційності ($a = \text{const}$ для всіх енергоблоків),
 γ_k – коефіцієнт, що враховує особливості роботи енергоблоку.

Обмеження на незалежні змінні N_i :

$$N_{\min i} \leq N_i \leq N_{\max i}$$

Обмеження типу рівності або рівняння зв'язку між N і N_k :

$$\sum_{k=1}^n N_k - N = 0$$

7. РЕАЛІЗАЦІЯ СТАТИЧНОЇ ОПТИМІЗАЦІЇ В СИСТЕМАХ КЕРУВАННЯ.

Для розв'язку завдань статичної оптимізації при керуванні технологічними процесами можуть застосовуватися наступні підходи:

- 1) Визначення оптимальних умов з використанням повної математичної моделі (керування по моделі).
- 2) Застосування спрощених математичних моделей з аналітичним розв'язком і використання методів розпізнавання ситуації для вибору моделі.
- 3) Розв'язок завдання оптимізації поза контуром керування й уточнення знайденого алгоритму в процесі функціонування системи залежно від величини збурень.
- 4) Розв'язок завдання статичної оптимізації за допомогою пошуку оптимуму безпосередньо на об'єкті керування.

Перераховані методи мають різні переваги і недоліки. Відзначимо, що перший метод припускає використання ефективної обчислювальної техніки, четвертий – найповнішу можливість спостереження, третій – пов'язаний із дуже великими витратами у підготовчій фазі.

Нижче будуть розглянуті стратегії розв'язку задачі статичної оптимізації за допомогою методів 1 та 2.

7.1. Оптимізація процесу за допомогою математичних моделей

У зв'язку із надзвичайною складністю повних математичних моделей, що описують сучасні виробництва, неможливо, як правило, знайти остаточний розв'язок завдання оптимізації. Необхідно прагнути до декомпозиції завдання оптимізації у зв'язку з тим, що з боку експлуатаційного персоналу підприємства висуваються вимоги до збільшення швидкодії й зменшенню обсягу пам'яті ЕОМ, зайнятий оптимізаційними алгоритмами і програмами.

7.1.1. Оптимізація на основі методів розпізнавання ситуації

Математичні моделі адекватно описують одну більш-менш вузьку область із загальної області можливої зміни керуючих впливів. Тому коефіцієнти моделі процесу повинні встановлюватися залежно від значень змінних стану та керування. Це завдання може бути частково вирішене за

допомогою алгоритмів розпізнавання ситуації, у тому випадку якщо відмовитися від уточнення моделі й обмежитися описом процесу по класах.

При цьому кожен клас стану процесу описується відповідною математичною моделлю.

Ситуація.

Під ситуацією розуміється сукупність всіх характеристик стану процесу на певний момент t ; характеристики складаються системою керування на основі інформації про процес і її обробки.

Розпізнавання ситуації.

Розпізнавання ситуації як методика ідентифікації вирішує два основні завдання:

- ✓ включення вектора стану процесу в обмежений, не порожній ряд натуральних або семантичних класів;
- ✓ розпізнавання й інтерпретація змін стану, які ведуть від бажаних умов проведення процесу до небажаного.

Оптимізацію процесу можна розділити на дві фази.

Попередня оптимізація (режим off-line).

Попередньо визначається оптимальне керування для різних класів стану. При цьому можуть застосовуватися будь-які як завгодно складні методи оптимізації. Оптимальні керуючі впливи зберігаються в запам'ятовувальному пристрої ЕОМ.

Оптимізація в ході технологічного процесу (режим on-line).

Відповідно до поточного стану знаходиться оптимальне керування. При цьому після визначення належності вектора стану до того або іншого класу здійснюється уточнення обраної моделі.

В умовах сучасного підприємства застосовні наступні стратегії оптимізації:

- А) Розв'язок повного завдання оптимізації по уточненій глобальній моделі.

До отримання результатів використовується оптимальне керування, отримане для представників різних класів.

- В) Ітеративне уточнення рішення повного завдання оптимізації.

Оптимальне керування отримане для представників різних класів використовується в якості вихідного.

- С) Розв'язок спрощеного завдання оптимізації для уточненої глобальної моделі.

Існують наступні варіанти спрощення, які можна комбінувати:

- ✓ спрощення цільової функції;
- ✓ зменшення розмірності вектора керування;
- ✓ зневага перешкодами.

Спрощення здійснюється на підставі результатів розв'язку завдання оптимізації для відповідних класів стану в режимі *off-line*. Результати розв'язку спрощеного завдання оптимізації повинні бути доповнені результатами оптимізації в режимі *off-line*.

D) Ітеративне наближення оптимального керування для уточненої глобальної моделі на основі спрощеної постановки завдання. Результати розв'язку завдання оптимізації для представників різних класів використовуються як вихідний розв'язок.

Застосування якогось із цих методів для оптимізації в режимі *on-line* залежить від конкретних умов, складності завдання оптимізації (виду цільової функції, складності математичної моделі, розмірності векторів стану і керування), а також від чутливості цільової функції до керуючих впливів.

Досвід показує, що за допомогою методів C і D можна значно скоротити час розрахунків, але в порівнянні з методами A і B практично завжди буде отримане менш ефективне керування. Вибір методу здійснюється залежно від можливостей обчислювальної техніки, необхідного часу досягнення екстремуму цільової функції і бажаних результатів оптимізації.

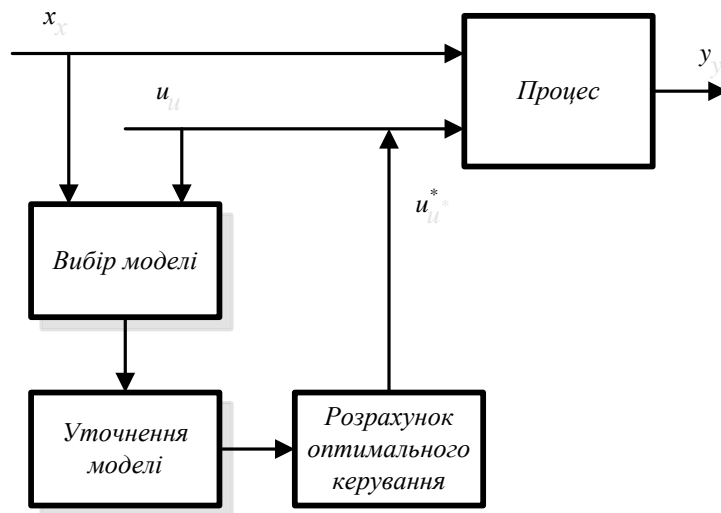


Рис. 7.1. Принципова схема попередньої оптимізації

Принципова схема попередньої оптимізації з математичною моделлю показана на рис. 7.1. З урахуванням вимірюваних величин збурень, вектору стану процесу $\vec{x} = |x_1, x_2, \dots, x_n|$ і вектору керування $\vec{u} = |u_1, u_2, \dots, u_s|$ здійснюється вибір моделі і її уточнення. Від операції вибору моделі можна відмовитися. У цьому випадку опис процесу здійснюється за допомогою однієї моделі, що підлягає уточненню.

Застосування методів розпізнавання ситуації пов'язане з наступними істотними перевагами:

- ✓ можливе сильне спрощення математичної моделі процесу;
- ✓ значне зниження часу реакції системи керування на збурення;
- ✓ підвищення надійності внаслідок модульної організації програмного забезпечення;
- ✓ знижена потреба в обсязі пам'яті ЕОМ.

Завдяки цим перевагам на практиці можлива реалізація досить складних систем керування технологічними процесами. Оптимальні значення керуючих впливів \vec{u}^* видаються автоматично або через оператора в систему керування технологічним процесом.

Характерним освоєнням цієї стратегії оптимізації є компенсація величин збурень до того, як вони почнуть впливати на технологічний процес.

7.1.2. Декомпозиційна оптимізація.

Декомпозиційна оптимізація заснована на розкладанні (декомпозиції) завдання оптимізації виробничого процесу. Зміст цього методу полягає в тому, що загальне завдання оптимізації наводять у вигляді ряду конкретних завдань. Вирішують ці завдання незалежно одне від одного і координують їх розв'язки таким чином, щоб задовольнити постановці загального завдання оптимізації (рис. 7.2).

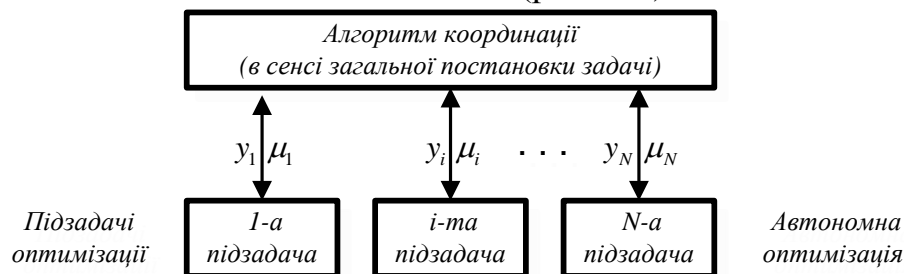


Рис. 7.2. Декомпозиційна оптимізація

Декомпозиційна оптимізація широко застосовується в керуванні технологічними процесами в основному з двох причин:

- для складних завдань оптимізації не існує одного надійного методу розв'язку;
- більша розмірність завдання оптимізації (до 15 і більше керуючих змінних) призводить до неприйняттого часу розв'язку завдання.

Крім того, знайдений розв'язок загального завдання може бути реалізовано за допомогою складних алгоритмів керування за допомогою ЕОМ, що, як правило, пов'язано з обмеженою надійністю системи.

Реалізація алгоритмів керування, що відповідають розв'язкам конкретних завдань, і їхня наступна координація зменшує можливість помилки.

Розглянемо наступну постановку завдання:

$$\max I = \max \sum_{i=1}^N F_i(\vec{x}_i, \vec{u}_i), \quad (7.1)$$

при умовах:

$$\vec{x}_i = \sum_{j=1}^N c_{ij} \cdot \vec{y}_j, \quad (7.2)$$

$$\vec{y}_j = \vec{f}_j(\vec{x}_j, \vec{u}_j), j = 1, 2, \dots, N. \quad (7.3)$$

Тут (рис. 7.3):

\vec{u}_i - вектор керування i -ї конкретної підсистеми:

$$\vec{u}_i = \{u_i^1, u_i^2, \dots, u_i^{\nu_i}\}; \quad (7.4)$$

\vec{x}_i - вектор вхідних величин i -ї конкретної підсистеми:

$$\vec{x}_i = \{x_i^1, x_i^2, \dots, x_i^{\alpha_i}\}; \quad (7.5)$$

\vec{y}_j - вектор вихідних величин j -ї конкретної підсистеми:

$$\vec{y}_j = \{y_j^1, y_j^2, \dots, y_j^{\beta_j}\}; \quad (7.6)$$

c_{ij} - матриця зв'язку.

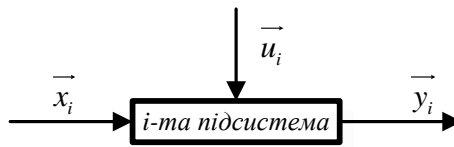


Рис. 7.3. Модель підсистеми

Необхідно оптимізувати N -східчасту систему, у якій кожен щабель впливає на загальну ефективність системи. Окремі конкретні підсистеми пов'язані між собою через (7.2), їхні математичні моделі описуються за допомогою (7.3).

Матриця зв'язку має розмірність $\alpha_i \times \beta_i$. Вектор вхідних величин у першу підсистему \vec{x}_1 - задається. Для завдання (7.1) - (7.3) може бути записана функція Лагранжа:

$$R(\vec{x}, \vec{y}, \vec{u}, \vec{\lambda}, \vec{\mu}) = \sum_{i=1}^N \{F_i(\vec{x}_i, \vec{u}_i) + \vec{\mu}_i^T (\sum_{j=1}^N c_{ij} \vec{y}_j - \vec{x}_i) + \vec{\lambda}_i^T [\vec{x}_i, \vec{u}_i] - \vec{y}_i\}. \quad (7.7)$$

Умови оптимальності мають вигляд:

$$\frac{\partial R}{\partial x_i} = \frac{\partial F_i}{\partial x_i} - \vec{\mu}_i + \vec{\lambda}_i \left(\frac{\partial f_i}{\partial x_i} \right)^T = 0; \quad (7.8)$$

$$\frac{\partial R}{\partial \bar{u}_i} = \frac{\partial F_i}{\partial u_i} + \bar{\lambda}_i \left(\frac{\partial \bar{f}_i}{\partial u_i} \right)^T = 0; \quad (7.9)$$

$$\frac{\partial R}{\partial \bar{y}_i} = -\bar{\lambda}_i + \sum_{j=1}^N c_{ij} \mu_j = 0; \quad (7.10)$$

$$\frac{\partial R}{\partial \bar{\lambda}_i} = \bar{f}_i - \bar{y}_i = 0; \quad (7.11)$$

$$\frac{\partial R}{\partial \bar{\mu}_i} = \sum_{j=1}^N c_{ij} \bar{y}_j - \bar{x} = 0. \quad (7.12)$$

Очевидно, що множники Лагранжа $\bar{\mu}_i$ і $\bar{\lambda}_i$ є векторами розмірності α_i (для $\bar{\mu}_i$) і β_i (для $\bar{\lambda}_i$).

Крім того:

$$\frac{\partial R}{\partial x_i} = \begin{pmatrix} \frac{\partial R}{\partial x_i^1} \\ \frac{\partial R}{\partial x_i^2} \\ \vdots \\ \frac{\partial R}{\partial x_i^{\alpha_i}} \end{pmatrix}; \quad (7.13)$$

$$\frac{\partial f_i}{\partial x_i} = \begin{pmatrix} \frac{\partial f_i^1}{\partial x_i^1} & \frac{\partial f_i^1}{\partial x_i^2} & \dots & \frac{\partial f_i^1}{\partial x_i^{\alpha_i}} \\ \frac{\partial f_i^2}{\partial x_i^1} & \frac{\partial f_i^2}{\partial x_i^2} & \dots & \frac{\partial f_i^2}{\partial x_i^{\alpha_i}} \\ \vdots & \vdots & & \vdots \\ \frac{\partial f_i^{\beta_i}}{\partial x_i^1} & \frac{\partial f_i^{\beta_i}}{\partial x_i^2} & \dots & \frac{\partial f_i^{\beta_i}}{\partial x_i^{\alpha_i}} \end{pmatrix}. \quad (7.14)$$

Отримати остаточний розв'язок завдання оптимізації практично неможливо через її складність. Необхідно відзначити, що (7.8), (7.9) і (7.11) містять лише змінні відповідної i -ї підсистеми. Ця особливість дозволяє (рис. 7.2) здійснити декомпозицію й знайти розв'язок шляхом координації розв'язків для окремих підсистем (конкретних задач). Суть координації полягає в тому, що різниці між оптимальними розв'язками взаємозалежних підсистем використовуються для корекції цих рішень. Вважається, що корекція рішень здійснюється на більше високому рівні керування, тобто на рівні керування всією технічною системою, а не на рівні окремих підсистем. У зв'язку із цим декомпозиційну оптимізацію називають також *багаторівневою* або *ієрархічною оптимізацією*.

Залежно від того, через які змінні здійснюється координація розв'язків для окремих підсистем, розрізняють два методи.

Метод І.

Координація здійснюється по множниках Лагранжа $\bar{\mu}_i$. Алгоритм розв'язку:

- 1) Вибір значень для множників Лагранжа: $\bar{\mu}_1, \bar{\mu}_2, \dots, \bar{\mu}_N$.
- 2) Визначення множників Лагранжа $\bar{\lambda}_1, \bar{\lambda}_2, \dots, \bar{\lambda}_N$ з (6.46).
- 3) Визначення величин \bar{x}_i, \bar{u}_i і \bar{y}_i із розв'язку системи рівнянь (7.8), (7.9), (7.11).
- 4) Уточнення $\bar{\mu}_i$ із застосуванням умови (6.48)

$$\bar{\mu}_i^{v+1} = \bar{\mu}_i^v + h \cdot \left(\sum_{j=1}^N c_{ij} \bar{y}_j - \bar{x}_i \right) \quad (7.15)$$

і перехід до п.2. Умовою закінчення процесу координації є виконання співвідношення:

$$\sum_{j=1}^N c_{ij} \bar{y}_j - \bar{x}_i \leq \bar{\varepsilon}, \quad (7.16)$$

де $\bar{\varepsilon}$ характеризує задану точність розв'язку. Величина h в (7.15) визначає довжину кроку. Розв'язок завдань оптимізації для окремих підсистем можна отримати за допомогою будь-яких методів. У п. 3 використовуються необхідні умови оптимальності. При фіксованих $\bar{\mu}_i, \bar{\lambda}_i$ можуть використовуватися пошукові методи. Великим недоліком цього методу є те, що він не дає допустимих розв'язків у змісті постановки завдання (7.1) – (7.3).

Метод ІІ.

Координація здійснюється по вихідних величинах \bar{y}_i . Алгоритм розв'язку:

- 1) Визначення значень вихідних величин $\bar{y}_1, \bar{y}_2, \dots, \bar{y}_N$.
- 2) Розв'язок системи рівнянь (7.8), (7.9), (7.11), (7.12) і знаходження величин $\bar{x}_i, \bar{u}_i, \bar{\mu}_i, \bar{\lambda}_i$.
- 3) Корекція \bar{y}_i з використанням (7.10)

$$\bar{y}_i^{v+1} = \bar{y}_i^v + h(-\bar{\lambda}_i + \sum_{j=1}^N c_{ij} \bar{\mu}_j) \quad (7.17)$$

і перехід до п. 2. Умовою закінчення процесу координації є виконання співвідношення:

$$-\bar{\lambda}_i + \sum_{j=1}^N c_{ij} \bar{\mu}_j \leq \bar{\varepsilon}. \quad (7.18)$$

На противагу методу І у цьому випадку отримують допустимий розв'язок, тому що рівняння (7.3) і співвідношення зв'язку (7.2) виконуються при кожному послідовному наближенні. Корекція відповідно до (7.17) поліпшує кожен проміжний розв'язок відповідно до цільової

функції (7.1). Розглянутий метод придатний для оптимізації в режимі *on-line*, тому що процес оптимізації може бути припинений після будь-якого послідовного наближення. Отримане в цей момент субоптимальний розв'язок є більш ефективним, ніж кожен знайдений раніше.

На закінчення необхідно вказати на одне обмеження. При визначенні \vec{y}_i з умови (7.12) визначаються також усі \vec{x}_i .

Умова (7.11) є системою β_i рівнянь для ν_i величини вектора \vec{u}_i . Наявність допустимого розв'язку для \vec{u}_i можливо лише тоді, коли виконується умова $\beta_i \leq \nu_i$. Інакше кажучи, метод II застосуємо для координації локальних завдань оптимізації лише тоді, коли розмірність вектору стану на виході кожної конкретної підсистеми не перевищує розмірності вектору керування відповідних підсистем.

7.2. Контрольні завдання

1. Мінімізувати функцію

$$f(x) = 40x_1 + 36x_2$$

при обмеженнях:

$$g_1(x) = x_1 - 8 \leq 0;$$

$$g_2(x) = x_2 - 10 \leq 0;$$

$$g_3(x) = 5x_1 + 3x_2 - 45 \geq 0;$$

$$g_4(x) = x_1 \geq 0;$$

$$g_5(x) = x_2 \geq 0.$$

2. Максимізувати функцію

$$f(x) = x_1 + 2x_2$$

при обмеженнях:

$$g_1(x) = x_1 + 2x_2 - 10 \leq 0;$$

$$g_2(x) = x_1 + x_2 - 1 \geq 0;$$

$$g_3(x) = x_2 - 4 \leq 0;$$

$$g_4(x) = x_1 \geq 0;$$

$$g_5(x) = x_2 \geq 0.$$

3. Максимізувати симплекс-методом функцію

$$f(x) = 3x_1 + 2x_2$$

при обмеженнях:

$$g_1(x) = -x_1 + 2x_2 - 4 \leq 0;$$

$$g_2(x) = 3x_1 + 2x_2 - 14 \leq 0;$$

$$g_3(x) = x_1 - x_2 - 3 \leq 0;$$

$$g_4(x) = x_1 \geq 0;$$

$$g_5(x) = x_2 \geq 0.$$

4. Максимізувати симплекс-методом функцію

$$f(x) = x_1 + 3x_2$$

при обмеженнях:

$$g_1(x) = x_1 - 5 \leq 0;$$

$$g_2(x) = x_1 + 2x_2 - 10 \leq 0;$$

$$g_3(x) = x_2 - 4 \leq 0;$$

$$g_4(x) = x_1 \geq 0;$$

$$g_5(x) = x_2 \geq 0.$$

Список використаної та рекомендованої літератури

1. Цирлин А.М. Оптимальное управление технологическими процессами – М.: Энергоатомиздат, 1986. – 400с.
2. Реклейтис Г. и др. Оптимизация в технике: В 2-х кн. Кн. 1. Пер. с англ. – М.: Мир, 1986. – 349 с.
3. Реклейтис Г. и др. Оптимизация в технике: В 2-х кн. Кн. 2. Пер. с англ. – М.: Мир, 1986. – 320 с.
4. Черноруцкий И.Г. Методы оптимизации в теории управления – СПб.: Питер, 2004. – 256 с.